DataArts Studio

User Guide

Issue 01

Date 2024-04-29

HUAWEI

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process.* For details about this process, visit the following web page:
https://www.huawei.com/en/psirt/vul-response-process
For vulnerability information, enterprise customers can visit the following web page:
https://securitybulletin.huawei.com/enterprise/en/security-advisory

Contents

1 DataArts Studio Introduction

DataArts Studio is a one-stop data operations platform that provides intelligent data lifecycle management. It supports intelligent construction of industrial knowledge libraries and incorporates data foundations such as big data storage, computing, and analysis engines. With DataArts Studio, your enterprise can easily construct end-to-end intelligent data systems. These systems can help eliminate data silos, unify data, accelerate data monetization, and promote digital transformation.

DataArts Studio Development Process

To use DataArts Studio, perform the following steps:

Table 1-1 DataArts Studio development process

Process	Description	Task	Helpful Link
Process design	Before using DataArts Studio, you are advised to analyze your business, clarify requirements, and design a process based on the capabilities provided by DataArts Studio. 1. Analyze requirements. Analyze your business, clarify requirements, and obtain the data governance framework to facilitate the design of a data governance process. 2. Conduct a survey. Determine the capability boundary of DataArts Studio and analyze the subsequent service load. 3. Design a process. Design the data governance process based on the business status and the capabilities of DataArts Studio. The process covers all the subsequent data governance operations.	1. Requirement analysis 2. Business survey 3. Process design	The process design is closely related to your business. You can design a process by referring to **Data Governance Based on Taxi Trip Data**. You can learn more by **contacting us**.
Preparations	If you access DataArts Studio for the first time, register an account with Huawei, buy a DataArts Studio instance, create a workspace and a user, authorize DataArts Studio permissions to the user, and add workspace members and roles.	Making preparations	**Preparations**
Management Center	Select cloud services for data storage, query, and analysis as required. Then, create data connections required for the cloud services.	Creating data connections	**Managing Data Connections**

Process	Description	Task	Helpful Link
DataArts Migration	Use DataArts Studio to upload data from data sources to the cloud. DataArts Migration migrates data between homogeneous and heterogeneous data sources such as self-built and cloud-based file systems, relational databases, data warehouses, NoSQLs, big data cloud services, and object storage.	Integrating data	**Supported Data Sources** **Creating a CDM Cluster** **Creating a Link** **Table/File Migration Jobs**
DataArts Catalog (metadata collection)	Collect metadata of raw data for data management and monitoring.	Collecting metadata	**Metadata Collection**
DataArts Architecture	Use DataArts Architecture to create entity-relationship (ER) models and dimensional models to standardize and visualize data development and output data governance methods that can guide development personnel to work with ease. In DataArts Architecture, you can create dimensions, fact tables, summary tables, and metrics that fit your needs.	Adding reviewers	**Adding a Reviewer**
		Managing Configuration Center	**Managing the Configuration Center**
		Designing processes	**Designing Processes**
		Designing subjects	**Designing Subjects**
		Managing lookup tables	**Creating Lookup Tables**
		Formulating data standards	**Creating Data Standards**
		Creating ER models	**ER Modeling**
		Dimensional modeling	**Dimensional Modeling**
		Business metrics	**Business Metrics**
		Technical metrics	**Technical Metrics**

Process	Description	Task	Helpful Link
		Data mart building	**Creating Summary Tables**
DataArts Factory	Use DataArts Factory to manage diverse big data services. The one-stop big data development environment enables a variety of operations such as data management, data integration, script development, job development, job scheduling, O&M, and monitoring, facilitating data analysis and processing.	Managing data	**Data Management Process**
		Developing scripts	**Script Development Process**
		Developing jobs	**Job Development Process**
		Performing O&M and scheduling	**Overview**
DataArts Quality	Use DataArts Quality to monitor business and technical metrics. Screen out unqualified data in a single column or cross columns, rows, and tables from the following perspectives: integrity, validity, timeliness, consistency, accuracy, and uniqueness. Use the automatically generated quality rules to standardize data repeatedly.	Monitoring business metrics	**Creating a Metric** **Creating a Rule** **Creating a Scenario**
		Monitoring data quality	**Creating Rule Templates** **Creating Quality Jobs** **Creating a Comparison Job**
DataArts Catalog (data map and permissions)	Use DataArts Studio DataArts Catalog to manage data permissions. DataArts Catalog provides data maps.	Data map	**Overview**
		Data permissions	**Overview**
DataArts DataService	Use DataArts DataService to centrally manage API services, create data APIs based on tables, and register APIs with DataArts DataService itself for unified management and publication.	Developing APIs	**Preparations** **Creating an API** **Debugging an API** **Publishing an API** **Managing APIs** **Creating Throttling Policies**

Process	Description	Task	Helpful Link
		Calling APIs	**Calling APIs**

2 Management Console

2.1 Tags

A tag is a key-value pair that identifies an instance. It consists of a key and a value.

DataArts Studio instance tags can be used in the following scenarios:

- If there are a large number of cloud resources, you can add tags to them (including DataArts Studio instances) by user, maintainer, or usage. Then you can use Tag Management Service (TMS) to identify tags and manage cloud resources easily.
- If there are multiple DataArts Studio instances, you can add tags to them by user, maintainer, or usage. Then you can search for and identify DataArts Studio instances by tag on the DataArts Studio instance list page.

Constraints

- A DataArts Studio instance can have a maximum of 20 tags.
- Each tag key must be unique. Only one tag value can be added to a tag key.

Procedure

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

Step 2 On the **Tags** page, click **Add/Edit Tag**. In the displayed dialog box, set parameters. Enter a tag key and a value, and click **Add**.

Figure 2-1 Adding/Editing tags

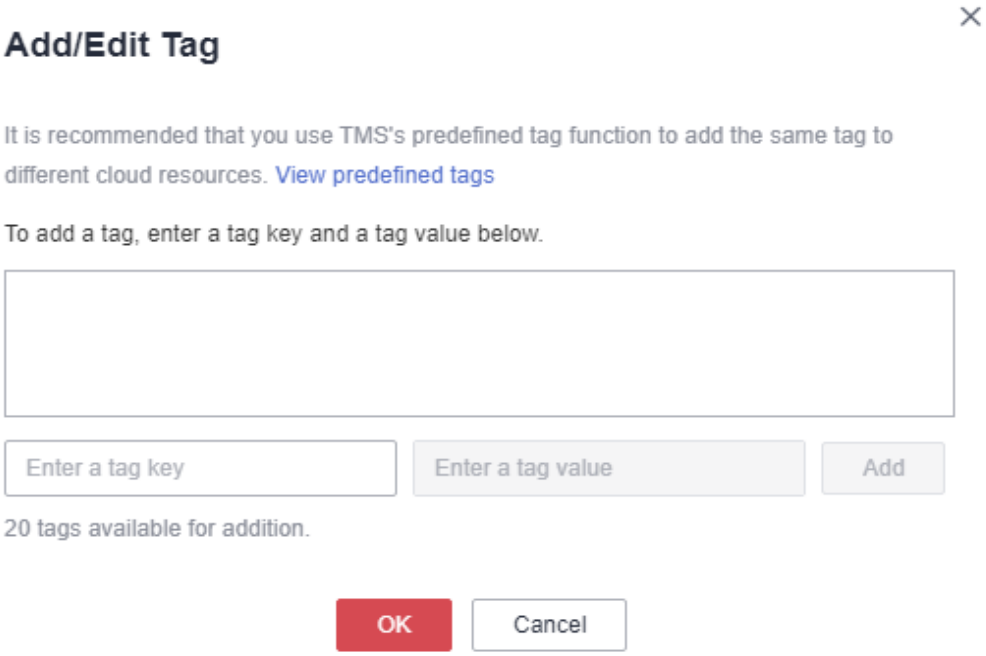


Table 2-1 Tag parameters

Parameter	Description
Tag Key	DataArts Studio supports predefined tags and resource tags. • Predefined tags are created on TMS. They may suit your needs to manage multiple services by tag. To use a predefined tag, you must create one on TMS first. Then you can select it from the **Tag key** drop-down list. You can click **View predefined tags** to enter the **Predefined Tag** page of TMS. Then, click **Create Tag** to create a predefined tag. For details, see **Creating Predefined Tags** in *Tag Management Service User Guide*. • Resource tags are created when you add a tag. You do not need to define them in advance. Resource tags are suitable for you to manage DataArts Studio instances by tag. To use a resource tag, you can enter the tag key in the box. The tag key can contain a maximum of 128 characters, including uppercase letters, lowercase letters, digits, spaces, and the following special characters: _.:=+-@. The tag key cannot start with a space or **_sys_**, or end with a space.

Parameter	Description
Tag Value	You can specify the tag value in either of the following ways: • Predefined tags: Click the text box and select a predefined tag value from the drop-down list. • Resource tags: Enter a tag value in the text box. The tag value can contain a maximum of 255 characters, including uppercase letters, lowercase letters, digits, spaces, and the following special characters: _.:=+-@. The tag value cannot start or end with a space.

----End

2.2 Enterprise Mode

2.2.1 DataArts Studio Enterprise Mode Overview

DataArts Studio provides two workspace modes, the simplified mode and enterprise mode, to help you manage your production data with varied security control requirements. This section describes the differences between the two modes from multiple dimensions, such as the physical form and impact on development.

NOTICE

Currently, only Management Center and DataArts Factory support the enterprise mode.

In simple mode, you need to create two workspaces, one for the development environment and the other for the production environment. In this way, you can isolate the development and production environment. You can export scripts or jobs from the development workspace and import them to the production workspace. In this mode, you cannot synchronize the production and development environment easily as there is no approval for the synchronization. To address these issues, you can use a workspace in enterprise mode to isolate the development and production environment. The one-click release and approval process improves your efficiency in task release.

You are advised to upgrade to the enterprise mode for your workspace to better manage the development process. For details, see **Creating a Workspace in Enterprise Mode**.

Background

This section contains the following parts which resolve the problems you may encounter when using the enterprise mode.

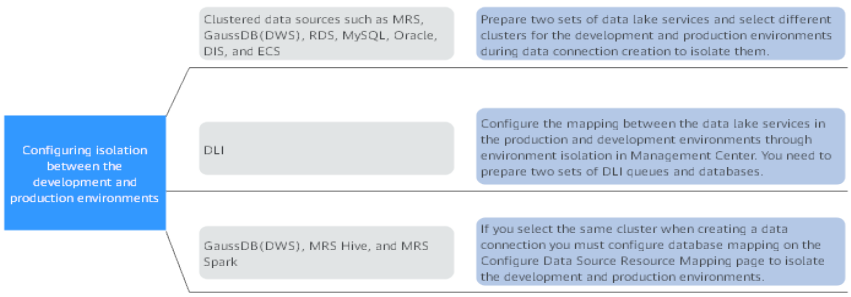
Table 2-2 Basics about the enterprise mode

Category	Description
Introduction to the Simple Mode and Enterprise Mode	Introduction to the two workspace modes
Comparison of Workspaces Using Different Modes in Production Task Development and O&M	Introduction to the task development and O&M mechanisms built based on the physical attributes of DataArts Studio workspaces
Advantages and Disadvantages of Workspace Modes	Comparison of the advantages and disadvantages of the workspace modes
Process of Using DataArts Studio in Different Workspace Modes	Process control of the workspace in enterprise mode
Operations Allowed by DataArts Studio Modules in Different Workspace Modes	In the simple mode, only the production environment is available. In the enterprise mode, the development environment and production environment are available. This part describes the mapping between environments and DataArts Studio modules.

Important Notes

- Different workspace modes have certain requirements on the data lake engine. To isolate the development environment from the production environment of a workspace that uses the enterprise mode, you must configure a data lake engine for both environments. You can configure isolation between the development and production environments using any of the methods shown in the following figure.

  Figure 2-2 Configuring isolation between the development and production environments

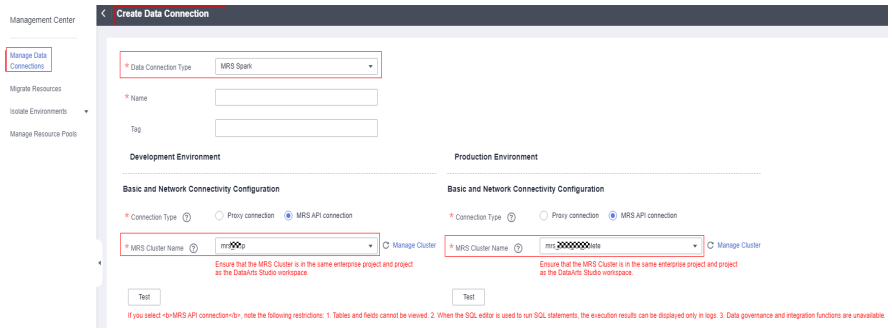
  

  - Configure two sets of data lake services to isolate the development environment from the production environment.

For clustered data sources, such as MRS, GaussDB(DWS), RDS, MySQL, Oracle, DIS, and ECS, you can create data connections in Management Center to distinguish data lake services in the development environment from those in the production environment. The data lake is automatically switched during development and production. Therefore, you need to prepare two sets of data lake services that have the same version, specifications, components, region, VPC, subnet, and other related configurations. For details, see **Creating a Data Connection**.

When creating a data connection, you can select different clusters for the development environment and production environment to isolate them.

Figure 2-3 Selecting different clusters during data connection creation



- Configure environment isolation for DLI.

  Configure environment isolation in enterprise mode, including DLI queue configuration and DB configuration.

  For serverless services (such as DLI), you can configure the mapping between data lake services in the production environment and those in the development environment through environment isolation in Management Center. The data lake is automatically switched during development and production. Therefore, you need to prepare two sets of queues and database resources in the serverless data lake service and distinguish them by name suffix. For details, see **Configuring Environment Isolation for a Workspace in Enterprise Mode**.

- Configure two databases in the same data lake service to isolate the development environment from the production environment.

  For GaussDB(DWS), MRS Hive, and MRS Spark, if you select the same cluster when creating a data connection (as shown in **Figure 2-4**), you must configure database mapping on the **Configure Data Source Resource Mapping** page shown in **Figure 2-5** to isolate the development and production environments. For details, see **DB configuration**.

Figure 2-4 Selecting the same cluster during data connection creation

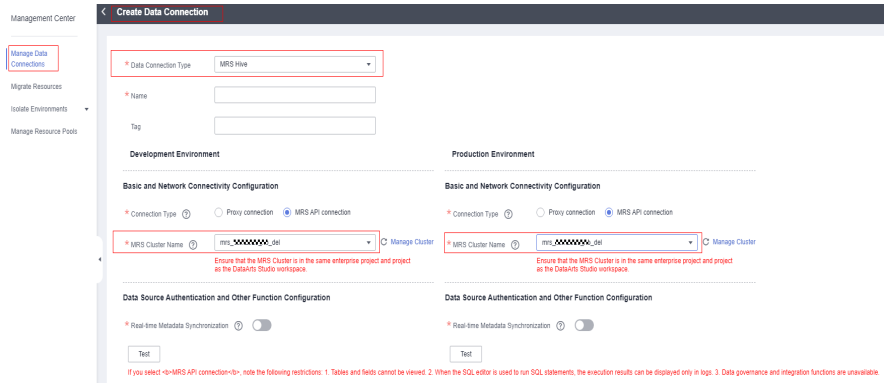


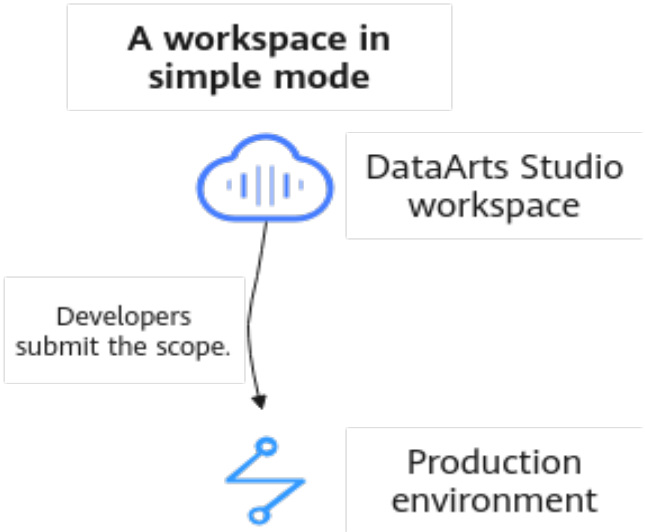
Figure 2-5 DB Configuration

- Data development jobs in the development environment of a workspace that uses the enterprise mode are not scheduled by default. They can be scheduled only after released to the production environment.

Introduction to the Simple Mode and Enterprise Mode

Typically, DataArts Studio workspaces use the simple mode. In this mode, you cannot isolate the development and production environment in the DataArts Factory and Management Center modules of DataArts Studio, or control the data development process or table permissions. Instead, you can only perform simple data development operations. A data lake functions as the production environment of DataArts Studio.

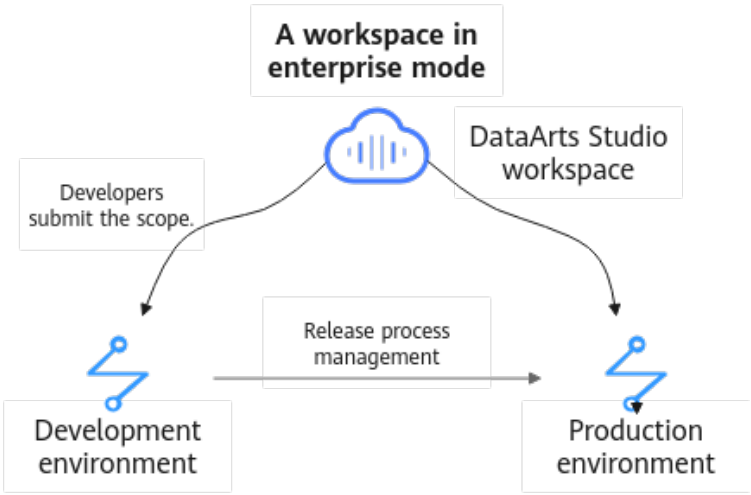
Figure 2-6 A workspace using the simple mode



The enterprise mode of DataArts Studio workspaces eliminates the risks of the simple mode. In this mode, you can isolate the development environment from the production environment in the DataArts Factory and Management Center modules of DataArts Studio. This prevents developers' operations from affecting services in the production environment. This mode requires two data lakes, one as the development environment and the other as the production environment.

- The development environment is accessible only to developers for script and job development and release of scripts and jobs to the production environment.
- The production environment is accessible only to end users and allows no change. Any change that is required must be made in the development environment and released to the production environment again.

Figure 2-7 A workspace using the enterprise mode



NOTE

- You can create a workspace in either mode to experience DataArts Studio. With a workspace in enterprise mode, you can isolate the code, compute resources, and permissions of the development environment from those of the production environment, and manage the task release process.
- If you are using a workspace in simple mode and want to experience the enterprise mode while retaining the code of the workspace, you can upgrade the workspace. For details, see **Creating a Workspace in Enterprise Mode**.

Comparison of Workspaces Using Different Modes in Production Task Development and O&M

Table 2-3 Comparison of workspaces using different modes in production task development and O&M

Comparison Item	Simple Mode	Enterprise Mode (Recommended)
Management of the production task development process	• After a task is submitted, it can be periodically executed to generate result data without being released. The process is submission and then production.	• You need to submit a task to the development environment and release the task to the production environment. Then the task can be automatically executed. The process is submission, release, and then production. • The production environment is accessible only to end users and allows no change. Any change that is required must be made in the development environment and released to the production environment again.
Management of the production task O&M permissions	Developers can directly edit scripts and jobs of production tasks.	Developers can edit and submit code on the DataArts Factory console, but cannot directly release code to the production environment. To release code to the production environment, developers must have the O&M permission. (The deployer, admin, and operator have this permission). • All scripts and jobs can be edited only in the development environment. The code in the production environment cannot be modified. • You can plan and manage task development and O&M processes on DataArts Studio based on the features of workspaces in enterprise mode and the role permission system of DataArts Studio. For details, see **Service Process in Enterprise Mode**.

Comparison Item	Simple Mode	Enterprise Mode (Recommended)
Management of production data permissions	Developers can directly use production data for tests, posing security threats to production data.	Developers can use test data in the development environment. Data in the production environment is read-only.

Advantages and Disadvantages of Workspace Modes

Table 2-4 Advantages and disadvantages of workspace modes

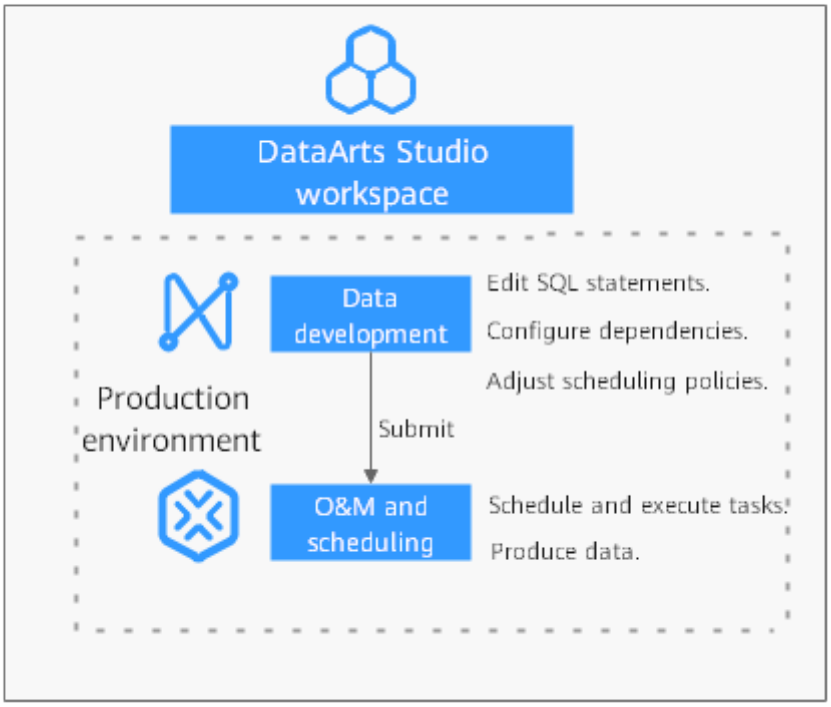
Comparison Item	Simple Mode	Enterprise Mode
Advantages	Simple, convenient, and easy to use • You only need to assign the developer role to data developers, and they are able to perform all data development tasks. • After submitting a script or job, you do not need to release it. The script or job can be periodically executed to generate result data.	Secure and normalized • A secure and normalized code release and management process (including code review and diff for checking code differences) is available. It ensures the stability of the production environment by avoiding unexpected circumstances such as dirty data spread and task errors caused by code logic. • Data access is effectively controlled to ensure data security. • All scripts and jobs can be edited only in the development environment. • Data in the development environment is isolated from that in the production environment. Developers cannot modify data in the production environment. • In the development environment, scripts and jobs are executed by the current developer. In the production environment, scripts and jobs are executed by a workspace-level public IAM account or public agency. • If any change is required for the production environment, the change must be made by a developer in the development environment first and then submitted to the production environment. The change can be successfully released only after being approved by the admin or deployer.

Comparison Item	Simple Mode	Enterprise Mode
Disadvantages	Unstable and insecure • The development environment cannot be isolated from the production environment. Only simple data development can be performed. • The permissions of production tables cannot be controlled. **NOTE** During development and commissioning, developers can directly access data in the production data lake and add, delete, and modify data in tables, posing threats to data security. • The data development process cannot be managed. **NOTE** Developers can add or modify scripts or jobs and submit them to the scheduling system without approval at any time, posing threats to service stability.	The process is relatively complex. Generally, one person cannot complete all data development and production tasks.

Process of Using DataArts Studio in Different Workspace Modes

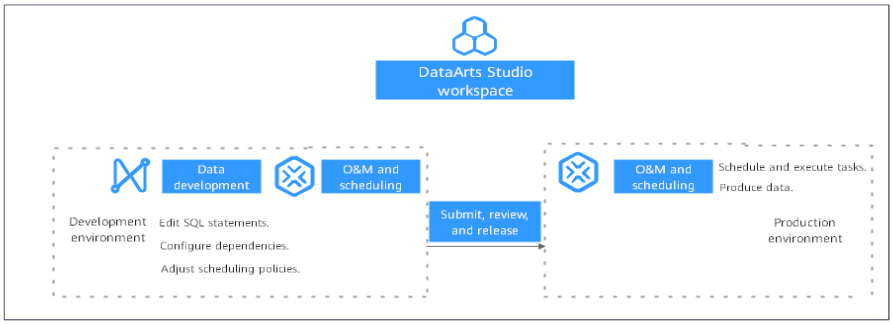
- In the simple mode, you cannot isolate the development and production environment in the DataArts Factory and Management Center modules of DataArts Studio, or control the data development process or table permissions. Instead, you can only perform simple data development operations. After submitting a script or job, you do not need to release it. The script or job can be periodically executed to generate result data.

Figure 2-8 Process in simple mode



- In the enterprise mode, you can isolate the development environment from the production environment in the DataArts Factory and Management Center modules of DataArts Studio. This prevents developers' operations from affecting services in the production environment. The development environment is accessible only to developers for script and job development and release of scripts and jobs to the production environment. The production environment is accessible only to end users and allows no change. Any change that is required must be made in the development environment and released to the production environment again.

Figure 2-9 Process in enterprise mode



Operations Allowed by DataArts Studio Modules in Different Workspace Modes

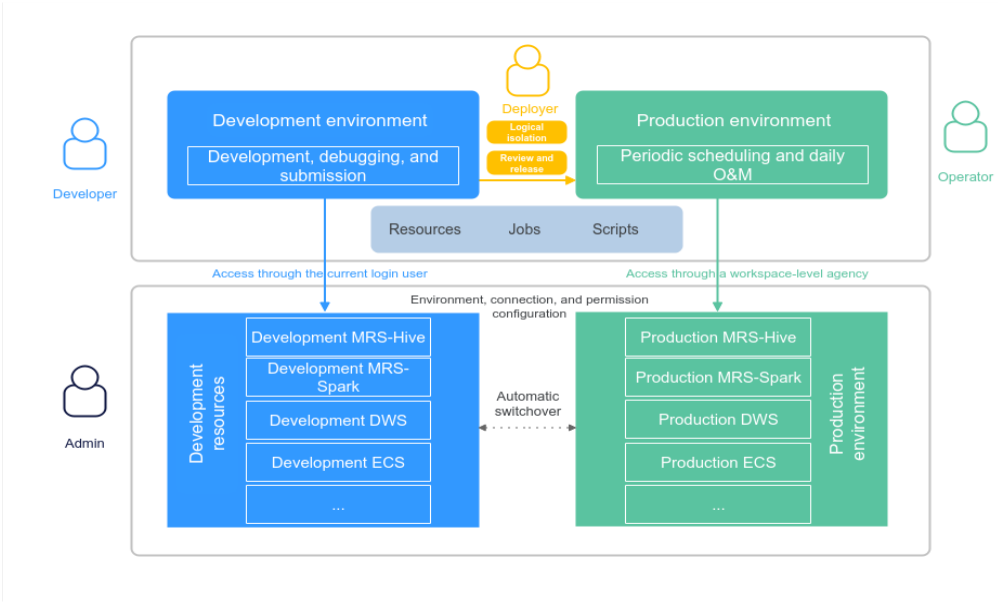
Table 2-5 Operations allowed by modules in different workspace modes

DataArts Studio Module	Simple Mode	Enterprise Mode
Management Center	Perform operations in the production environment (data connection operations and data import and export).	Perform operations in the development and production environments (data source resource mapping configuration, data connection operations, and data import and export)
DataArts Factory	Perform operations on instances and databases in the production environment.	Perform operations on instances and databases in the development and production environments.

2.2.2 Service Process in Enterprise Mode

The DataArts Studio enterprise mode mainly involves the Management Center and DataArts Factory components. The service process is completed by the admin, developer, deployer, and operator.

Figure 2-10 Enterprise mode architecture



- The admin performs operations such as preparing a data lake, configuring data connections and environment isolation, importing and exporting data, and configuring project user permissions.
- The developer develops and tests scripts and jobs, and submits versions and release tasks in the development environment.
- The deployer reviews the submitted tasks in the development environment.
- The operator performs operations such as job monitoring, notification management, and backup in the production environment based on the resources released by the developer.
- Custom role: You can customize operation permissions to meet your requirements.
- The viewer can only read data from DataArts Studio, but cannot perform operations or modify work items or configurations. You are advised to assign this role to users who only want to view information in the workspace.

Table 2-6 Permissions of different roles

Role	Simple Workspace	Enterprise Workspace
Admin	Has all permissions of Management Center in the production environment, including connection configuration and data import and export.	• Deployment-related operations • Connection configuration, environment isolation configuration, and data import and export in Management Center • Configuration in DataArts Factory, such as the environment, scheduling identity, and default item configuration
Developer	Has all permissions to develop jobs and scripts in the production environment.	• Development environment: all permissions • Production environment: read-only permission • Deployment: packaging and viewing release items, viewing the release item list, and viewing the release package content • Environment information configuration: read-only permission
Deployer	None	• Viewing release packages • Viewing the release item list • Releasing packages: Only the deployer and admin can perform this operation. • Canceling a release: Only the deployer and admin can perform this operation.

Role	Simple Workspace	Enterprise Workspace
Operator	Has the permissions to monitor, schedule, and perform O&M operations on the job and script instances in the production environment.	• Development environment: read-only permission • Production environment: all permissions • Deployment: viewing the release package content • Environment information configuration: read-only permission
Viewer	Read-only permission	Read-only permission

2.2.3 Creating a Workspace in Enterprise Mode

If you are using a workspace in simple mode and want to isolate the development and production environments, you can upgrade the workspace to one in enterprise mode. If you have not used any workspace in simple mode and do not need to inherit data, you can directly create a workspace in enterprise mode by following the instructions in this section.

Restrictions

You can upgrade your workspace mode or create a workspace in enterprise mode only if you are assigned the DAYU Administrator or Tenant Administrator role.

Prerequisites

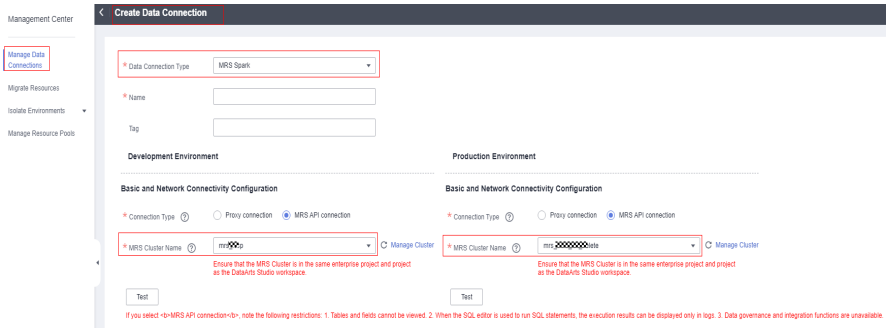
Before creating a workspace, ensure that:

- You have understood the differences between workspaces in simple mode and those in enterprise mode, such as the differences in the development process. For details, see **Introduction to the Simple Mode and Enterprise Mode**.
- You have configured workspace-level scheduling identities, including public agencies and public IAM accounts. For details, see **Configuring a Public Agency** and **Configuring a Public IAM Account**.
- You have prepared two sets of isolated data lake engines, one for the development environment and the other for the production environment.
  - Configure two sets of data lake services to isolate the development environment from the production environment.

    For clustered data sources, such as MRS, GaussDB(DWS), RDS, MySQL, Oracle, DIS, and ECS, you can create data connections in Management Center to distinguish data lake services in the development environment from those in the production environment. The data lake is automatically switched during development and production. Therefore, you need to prepare two sets of data lake services that have the same version, specifications, components, region, VPC, subnet, and other related configurations. For details, see **Creating a Data Connection**.

    When creating a data connection, you can select different clusters for the development environment and production environment to isolate them.

Figure 2-11 Selecting different clusters during data connection creation



– Configure environment isolation for DLI.

  Configure environment isolation in enterprise mode, including DLI queue configuration and DB configuration.

  For serverless services (such as DLI), you can configure the mapping between data lake services in the production environment and those in the development environment through environment isolation in Management Center. The data lake is automatically switched during development and production. Therefore, you need to prepare two sets of queues and database resources in the serverless data lake service and distinguish them by name suffix. For details, see **Configuring Environment Isolation for a Workspace in Enterprise Mode**.

– Configure two databases in the same data lake service to isolate the development environment from the production environment.

  For GaussDB(DWS), MRS Hive, and MRS Spark, if you select the same cluster when creating a data connection (as shown in **Figure 2-12**), you must configure database mapping on the **Configure Data Source Resource Mapping** page shown in **Figure 2-13** to isolate the development and production environments. For details, see **DB configuration**.

Figure 2-12 Selecting the same cluster during data connection creation

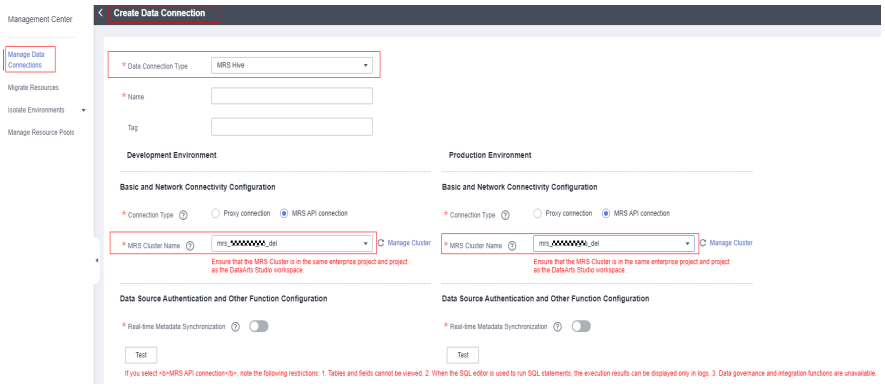


Figure 2-13 DB Configuration

- Prepare and synchronize data.
  - After creating data lake services, you must create databases, database schemas (required only for DWS), and data tables in the data lake services of the development and production environments based on the project plan (for example, the databases and tables required for data development).
    - For clustered data sources (such as MRS, DWS, RDS, MySQL, Oracle, DIS and ECS), use two clusters, one for the development environment and the other for the production environment. The names of the databases, database schemas (required only for DWS), and data tables in the two environments must be the same.
    - For serverless services (such as DLI), you are advised to associate and distinguish the two queues and databases by name suffix (add suffix **_dev** to the names of the queues and databases in the development environment and add no suffix to those in the production environment). The names of data tables in the development environment must be the same as those in the production environment.
    - For DWS, MRS Hive, and MRS Spark data sources, if the same cluster is used for the development and production environments, use two databases to isolate the development and production environments (add suffix **_dev** to the database for the development environment and add no suffix to the database for the production environment). The names of database schemas (required only for DWS) and data tables in the development environment must be the same as those in the production environment.
  - After creating databases, database schemas (required only for DWS), and data tables, you must synchronize data of existing tables (if any) between the two data lake services.
    - Existing data in data lakes: Use data migration services such as CDM and DRS to synchronize data in batches between data lakes.
    - Data to be migrated from the data source: Use peering jobs of data migration services such as CDM and DRS to synchronize data between the data lake service of the production environment and that of the development environment.

Change Description

After the workspace mode is upgraded, a development environment isolated from the production environment is added.

Upgrading the Simple Mode to Enterprise Mode

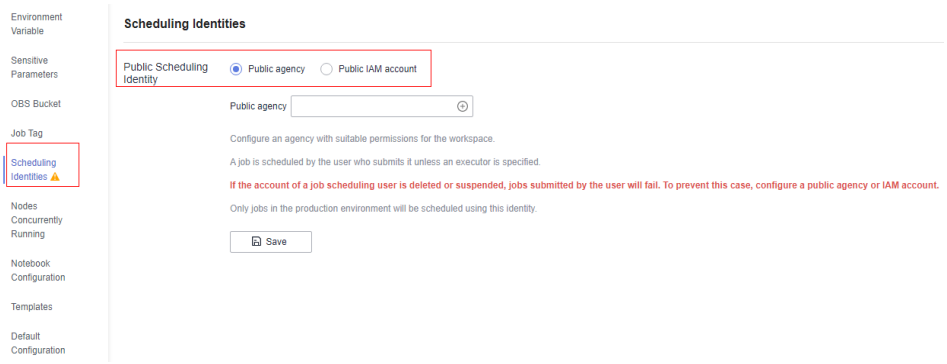
With the DAYU Administrator or Tenant Administrator role, you can upgrade a workspace in simple mode to one in enterprise mode.

- Pre-upgrade operations

  Configure a workspace-level public agency or public IAM account in DataArts Factory to avoid an upgrade failure.

  For details about how to configure an agency, see **Configuring a Scheduling Identity**.

  Figure 2-14 Configuring a workspace-level agency



- Upgrade operations

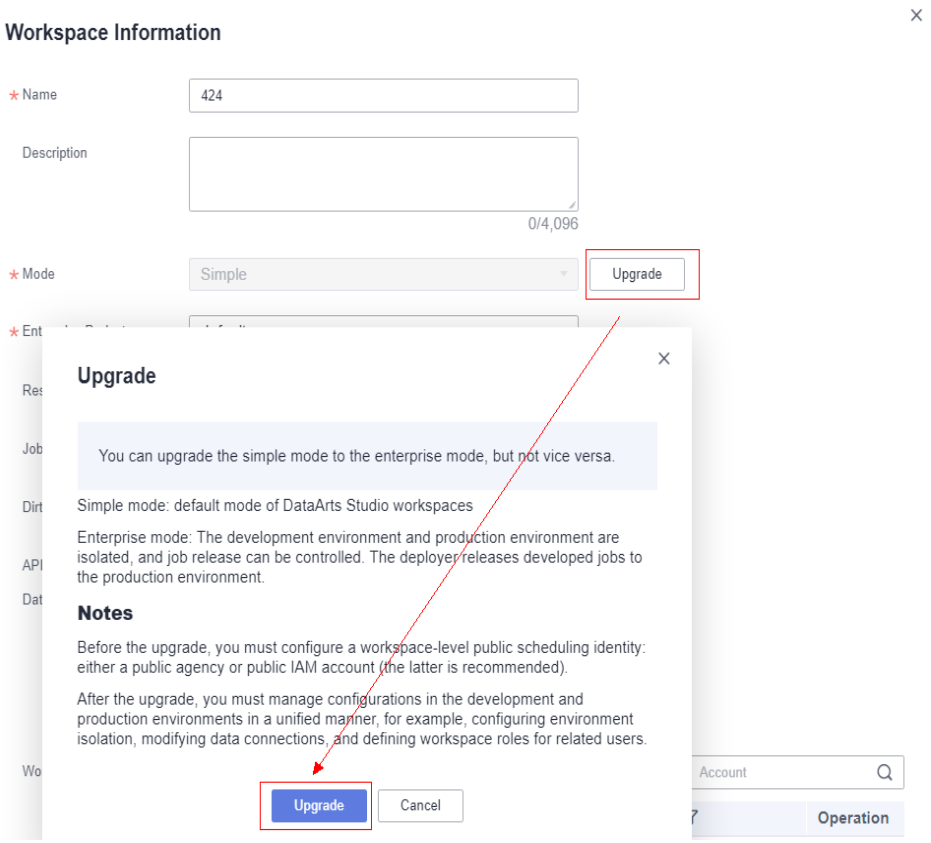
  a. Log in to the DataArts Studio console.

  b. Locate a DataArts Studio instance and click **Access**. Then, click the **Workspaces** tab.

  c. Locate the workspace you want to upgrade and click **Edit** in **Operation** column.

  d. In the displayed **Workspace Information** dialog box, click **Upgrade** next to the **Mode** text box. In the displayed dialog box, click **Upgrade**.

Figure 2-15 Upgrading to the enterprise mode



- Post-upgrade operations

  After the upgrade is complete, you (as the admin) need to modify data connections, configure environment isolation, and define roles such as the admin, developer, deployer, and operator in the workspace.

  a. Modify data connections. For details, see **Creating a Data Connection**.

  b. Configure environment isolation. For details, see **Configuring Environment Isolation for a Workspace in Enterprise Mode**.

  c. Define workspace roles for other users: For details, see **Adding a Member and Assigning a Role**.

Creating a Workspace in Enterprise Mode

If you have not used the simple mode before and do not need to inherit business data, you can directly create a workspace in enterprise mode.

- Create a workspace.

  a. Log in to the DataArts Studio console using an account with the DAYU Administrator or Tenant Administrator permission.

  b. Locate an instance and click **Access**. Then click the **Workspaces** tab.

  c. Click **Create**. In the displayed **Create** dialog box, set parameters based on **Table 2-7** and click **OK**.

Figure 2-16 Creating a workspace

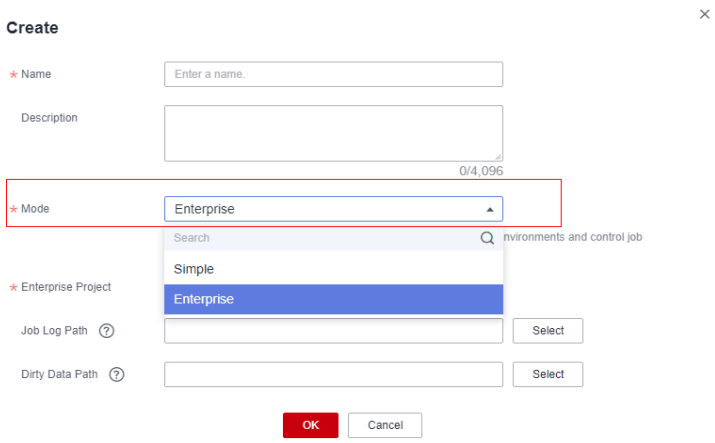


Table 2-7 Parameters for creating a workspace

Parameter	Description
Name	Workspace name. It can contain a maximum of 32 characters, including only letters, digits, underscores (_), and hyphens (-). The workspace name must be unique in the current DataArts Studio instance.
Description	Workspace description
Mode	Mode of the workspace. Available options include **Simple** and **Enterprise**. Select **Enterprise**.
Enterprise Project	Enterprise project associated with the default workspace of the DataArts Studio instance. An enterprise project facilitates management of cloud resources. For details, see **Enterprise Management User Guide**. This parameter is available only when an enterprise project has been created. If you want to connect the DataArts Studio instance to a cloud service (such as DWS, MRS, and RDS), ensure that the enterprise project of the DataArts Studio workspace is the same as that of the cloud service instance. • You can buy only one DataArts Studio instance for an enterprise project. • If you want to enable communication between DataArts Studio and another cloud service, ensure that the enterprise project of DataArts Studio is the same as that of the cloud service.

Parameter	Description
Job Log Path	OBS bucket for storing the job logs of DataArts Factory of DataArts Studio. To use the DataArts Factory module of DataArts Studio, workspace members must have the read and write permissions on the OBS bucket for storing job logs. Otherwise, the system cannot read or write job logs of DataArts Factory. • Click **Select**. You can select an existing OBS bucket. The selected OBS bucket is globally configured in the current workspace. • If you do not set this parameter, job logs of DataArts Factory are stored in the OBS bucket named **dlf-log-{projectId}** by default.
Dirty Data Path	OBS bucket for storing dirty data generated during DLI SQL execution in DataArts Factory of DataArts Studio. To use DataArts Factory to develop and execute DLI SQL statements, workspace members must have the read and write permissions on the OBS bucket where DLI dirty data is stored. Otherwise, the system cannot read or write the dirty data generated during DLI SQL execution. • Click **Select**. You can select a created OBS bucket. The selected OBS bucket is globally configured in the current workspace. • If you do not set this parameter, dirty data generated during DLI SQL execution is stored in the OBS bucket named **dlf-log-{projectId}** by default.
API Quota of DataArts DataService Exclusive	The value of this parameter indicates the used quota, allocated quota, total used quota, total allocated quota, or total quota, respectively. If you use DLM Exclusive, you will be charged for the APIs you have created. The default quota is 0, which means that no APIs can be created. The initial workspace has a trial quota of 10 APIs. The allocated quota can be modified. The allocated quota cannot be less than the used quota or greater than the sum of the total quota and the previously allocated quota minus the total allocated quota.

- Perform follow-up operations.

  After creating the workspace, you (as the admin) need to create data connections, configure environment isolation, and define roles such as the admin, developer, deployer, and operator in the workspace.

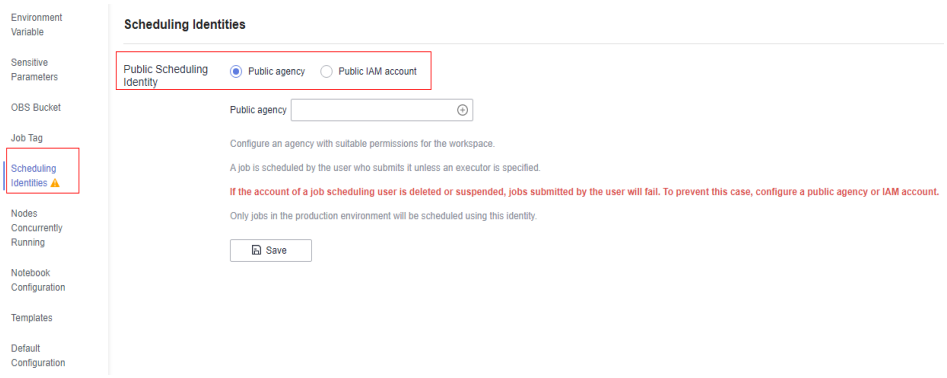
  a. Create data connections. For details, see **Creating a Data Connection**.

  b. Configure environment isolation. For details, see **Configuring Environment Isolation for a Workspace in Enterprise Mode**.

c. Define workspace roles for other users: For details, see **Adding a Member and Assigning a Role**.

d. Configure a workspace-level public agency or public IAM account in DataArts Factory. For details about how to configure an agency, see **Configuring a Scheduling Identity**.

Figure 2-17 Configuring a workspace-level agency



2.2.4 Admin Operations

As the project owner or development owner, the admin manages the environment configuration and personnel roles in enterprise mode in a unified manner. The following table describes related operations.

Table 2-8 Admin operations

Operation	Description
Making preparations	The preparations include preparing data lakes and preparing and synchronizing data. **Preparing data lakes:** In enterprise mode, the development environment and production environment need to be isolated. Therefore, you need to prepare two data lake services, one for the production environment and the other for the development environment. ● For clustered data sources, such as MRS, GaussDB(DWS), RDS, MySQL, Oracle, DIS, and ECS, you can create data connections in Management Center to distinguish data lake services in the development environment from those in the production environment. The data lake is automatically switched during development and production. Therefore, you need to prepare **two data lake services (clusters)** that have the same version, specifications, components, region, VPC, subnet, and other related configurations. For example, if your data lake service is an MRS cluster, you need to prepare two MRS clusters with the same version, specifications, components, region, VPC, and subnet. Any change to the configuration of one of the MRS clusters must be synchronized to the other cluster. ● For serverless services (such as DLI), you can configure the mapping between data lake services in the production environment and those in the development environment through environment isolation in Management Center. The data lake is automatically switched during development and production. Therefore, you need to prepare two sets of queue and database resources in the serverless data lake service. You are advised to distinguish them by name suffix. ● If GaussDB(DWS), MRS Hive, and MRS Spark data sources use **the same cluster**, you must configure database mapping on the **Configure Data Source Resource Mapping** page to isolate the development and production environments.

Operation	Description
	Preparing and synchronizing data: ● After creating data lake services, you must create databases, database schemas (required only for DWS), and data tables in the data lake services of the development and production environments based on the project plan (for example, the databases and tables required for data development). – For clustered data sources (such as MRS, DWS, RDS, MySQL, Oracle, DIS and ECS), use two clusters, one for the development environment and the other for the production environment. The names of the databases, database schemas (required only for DWS), and data tables in the two environments must be the same. – For serverless services (such as DLI), you are advised to associate and distinguish the two queues and databases by name suffix (add suffix _**dev** to the names of the queues and databases in the development environment and add no suffix to those in the production environment). The names of data tables in the development environment must be the same as those in the production environment. – For DWS, MRS Hive, and MRS Spark data sources that use the same cluster, use two databases to isolate the development and production environments (add suffix _**dev** to the database for the development environment and add no suffix to the database for the production environment). The names of database schemas (required only for DWS) and data tables in the development environment must be the same as those in the production environment. ● After creating databases, database schemas (required only for DWS), and data tables, you must synchronize data of existing tables (if any) between the two data lake services. – Existing data in data lakes: Use data migration services such as CDM and DRS to synchronize data in batches between data lakes.

Operation	Description
	– Data to be migrated from the data source: Use peering jobs of data migration services such as CDM and DRS to synchronize data between the data lake service of the production environment and that of the development environment.
Creating data connections in enterprise mode	You must create data connections for all data lake engines. For clustered data sources that use different clusters, you can create a data connection between DataArts Studio and the data lake of the development environment and a data connection between DataArts Studio and the data lake of the production environment at the same time. For details, see **Creating a Data Connection**.
Configuring environment isolation for a workspace in enterprise mode	Configure DLI queue and DB mapping to isolate the development and production environments. For the DWS, MRS Hive, and MRS Spark data sources, if you select the same cluster when creating a data connection, you need to configure two databases for the same data lake service to isolate the development environment from the production environment. For details, see **DB Configuration**. For the DLI data source, you can configure two DLI queues and databases to isolate the production environment from the development environment. For details, see **Configuring Environment Isolation for a Workspace in Enterprise Mode**.
Creating IAM users and assigning DataArts Studio permissions	Create IAM accounts with the **DAYU User** permission for project members who use DataArts Studio. For details, see **Creating an IAM User and Assigning DataArts Studio Permissions**.
Adding workspace members and assigning roles	Assign workspace roles to the IAM accounts of the project members. Six types of roles are available: admin, developer, deployer, operator, viewer, and custom role. For details, see **Adding a Member and Assigning a Role**.

2.2.5 Developer Operations

The developer develops scripts and jobs. The following table describes related operations.

Table 2-9 Developer operations

Operation	Description
Script development	Select the data lake engine for the development environment, and commission and release data development scripts in the development environment. After the release, the engine is automatically replaced by the engine of the production environment. For details, see **Script Development**.
Job development	Select the data lake engine for the development environment, and commission and release data development jobs in the development environment. After the release, the engine is automatically replaced by the engine of the production environment. For details, see **Job Development**.

2.2.6 Deployer Operations

- The deployer reviews the tasks to be released. This section describes related operations.
- The deployer reviews the release tasks submitted by the developer. Modified jobs can be synchronized to the production environment only after the corresponding release tasks are approved.

In enterprise mode, when a developer submits a script or job version, the system generates a release task. After the developer confirms the release and the deployer approves the release request, the modified job is synchronized to the production environment.

Prerequisites

The developer has completed the operations in **Releasing a Script Task** or **Releasing a Job Task**.

Procedure

Step 1 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

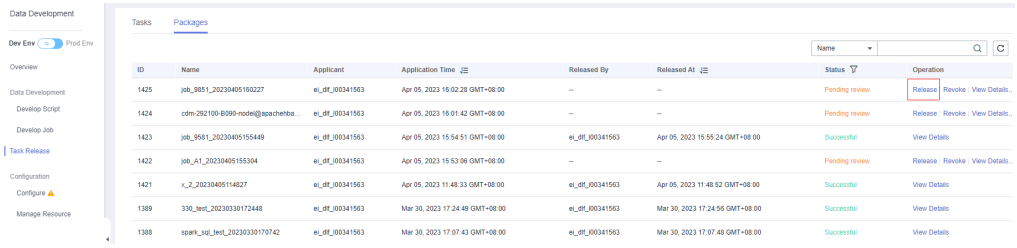
Step 2 In the left navigation pane, choose **Data Development** > **Task Release**.

Step 3 Click the **Packages** tab. You can click **View Details** in the **Operation** column to view the changes of the task compared with its previous version.

- If there is any issue, click **Revoke** to reject the release task. After the developer modifies and submits the release task again, you can review it again.

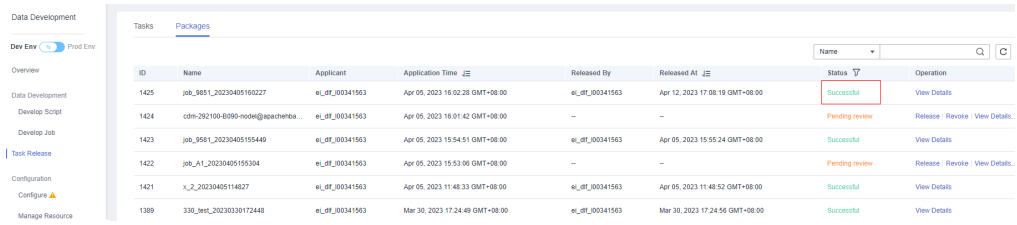
- After confirming that the release task has no remaining issue, click **Release** to approve the task.

Figure 2-18 Reviewing and releasing a task



Step 4 After the task is released, you can view its status. The developer's modification is synchronized to the production environment.

Figure 2-19 Viewing the task status



----End

2.2.7 Operator Operations

The operator manages the jobs, instances, notifications, and backups in the production environment in a unified manner. The following table describes related operations.

Table 2-10 Operator operations

Operation	Description
Job monitoring	Monitor batch and real-time jobs. For details, see **Monitoring a Job**.
Instance monitoring	Monitor job instances (a job instance is generated each time a job is executed). For details, see **Instance Monitoring**.
PatchData monitoring	Monitor the statuses of PatchData jobs. PatchData can be used to fix the job instances that have data errors in the historical records or to build job records for debugging programs. For details, see **Monitoring PatchData**.

Operation	Description
Notification management	Configure notifications to be sent when a job is abnormal or runs successfully. For details, see **Managing Notifications**.
Backup management	Back up all the jobs, scripts, resources, and environment variables of the previous day at a specified time on each day. For details, see **Managing Backups**.

3 Management Center

DataArts Studio Management Center provides a unified configuration and management entry for data connections and resource migration. Personalized entries and showcases can be customized as needed.

3.1 Data Sources

Before using DataArts Studio, you need to select cloud services or databases as the data lake foundation, which provides storage and compute capabilities. DataArts Studio provides one-stop data development, governance, and services based on the data lake foundation.

Data Sources Supported By DataArts Studio

DataArts Studio can interconnect with cloud services such as DWS, DLI, and MRS Hive as well as traditional databases such as MySQL and Oracle. For details, see **Table 3-1**.

To connect to these data sources, go to the DataArts Studio console and choose **Management Center** to create a data connection.

NOTE

The data connections in DataArts Studio Management Center are used to connect to the data lake foundation. DataArts Studio provides one-stop data development, governance, and services based on the data lake foundation.

Table 3-1 Data sources supported by DataArts Studio

Data Source Type	Management Center	DataArts Architecture	DataArts Factory	DataArts Catalog[2]	DataArts Quality[3]	DataArts DataService
DWS	Supported	Supported	Supported	Supported	Supported	Supported

Data Source Type	Management Center	DataArts Architecture	DataArts Factory	DataArts Catalog[2]	DataArts Quality[3]	DataArts DataService
DLI	Supported	Supported	Supported	Supported	Supported	Supported
MRS HBase	Supported	Not supported	Not supported	Supported	Not supported	×
MRS Hive	Supported	Supported	Supported	Supported	Supported	Not supported
MRS Kafka	Supported	Not supported	Supported	Not supported	Not supported	Not supported
MRS Spark[1]	Supported	Supported	Supported	Not supported	Supported	Not supported
MRS ClickHouse	Supported	Supported	Supported	Supported	Not supported	Supported
MRS Hetu	Supported	Not supported	Supported	Not supported	Supported	Supported
MRS Impala	Supported	Not supported	Supported	Not supported	Not supported	Not supported
MRS Ranger	Supported	Not supported	Not supported	Not supported	Not supported	Not supported
MapReduce (MRS) Presto	Supported	Not supported	Supported	Not supported	Not supported	Not supported
MRS Doris	Supported	Supported	Supported	Supported	Not supported	Supported
RDS for MySQL	Supported	Supported	Supported	Supported	Supported	Supported
RDS for PostgreSQL	Supported	Supported	Supported	Supported	Supported	Not supported

Data Source Type	Management Center	DataArts Architecture	DataArts Factory	DataArts Catalog[2]	DataArts Quality[3]	DataArts DataService
MySQL	Supported	Supported	Not supported	Not supported	Supported	Supported
Oracle	Supported	Supported	Not supported	Supported	Supported	Not supported
Data Ingestion Service (DIS)	Supported	Not supported	Supported	Supported	Not supported	Not supported
Host Connection	Supported	Not supported	Supported	Not supported	Not supported	Not supported

Annotation

[1] DataArts Catalog: In addition to the data sources listed in the preceding table, DataArts Catalog can also collect metadata of the following data sources:

1. Relational databases, such as MySQL and PostgreSQL databases (You can use RDS connections to collect the metadata of these databases.)
2. Cloud Search Service (CSS)
3. Graph Engine Service (GES)
4. Object Storage Service (OBS)
5. MRS Hudi (MRS Hudi is a data format. The metadata is stored in Hive, and operations are performed using Spark.) You can enable synchronization of the Hive table configuration for Hudi tables, and then you can collect the metadata of Hudi tables by collecting the MRS Hive metadata.

[2] The quality jobs and comparison jobs of DataArts Quality are not supported by MRS clusters with decoupled storage and compute.

[3] MRS Spark: MRS Spark connections can be used to integrate data into the DataArts Architecture and DataArts Quality modules. MRS Hudi is a data format. The metadata is stored in Hive, and operations are performed using Spark. DataArts Catalog uses MRS Hive to collect Hudi metadata, and DataArts Architecture and DataArts Quality use MRS Spark to govern Hudi data sources. (Business metric monitoring of DataArts Quality does not support Hudi data sources.)

Overview

Table 3-2 Data source overview

Data Source Type	Description
DWS	HUAWEI CLOUD DWS employs the shared-nothing architecture and massively parallel processing (MPP) engine. It is compatible with ANSI SQL 99, SQL 2003, and the PostgreSQL or Oracle database ecosystem, providing competitive solutions for analyzing petabytes of data in various industries.
DLI	HUAWEI CLOUD DLI is a serverless big data compute and analysis service that is fully compatible with Apache Spark and Apache Flink ecosystems. With multi-model engines supported by DLI, enterprises can use SQL statements or programs to easily complete batch processing, stream processing, in-memory computing, and machine learning of heterogeneous data sources.
MRS HBase	HBase undertakes data storage. It is an open-source, column-oriented, distributed storage system that is suitable for storing massive amounts of unstructured or semi-structured data. It features high reliability, high performance, and flexible scalability, and supports real-time data read/write. MRS HBase stores massive amount of data and supports data queries in milliseconds. MRS HBase can load and update logistics data in milliseconds, and query and analyze petabytes of time series data in seconds.
MRS Hive	Hive is a mechanism that can store, query, and analyze large-scale data stored in Hadoop. Hive defines simple SQL-like query language, which is known as HiveQL. It allows users familiar with SQL to query data. MRS Hive can be used to analyze terabytes or petabytes of data and quickly migrate on-premises Hadoop big data platforms (such as CDH and HDP) to the cloud without service interruption and service code modification.

Data Source Type	Description
MRS Kafka	HUAWEI CLOUD MRS provides dedicated MRS Kafka clusters. Kafka is an open-source, distributed, partitioned, and replicated commit log service. Kafka is publish-subscribe messaging, rethought as a distributed commit log. It provides features similar to Java Message Service (JMS) but another design. It features message endurance, high throughput, distributed methods, multi-client support, and real time. It applies to both online and offline message consumption, such as regular message collection, website activeness tracking, aggregation of statistical system operation data (monitoring data), and log collection. These scenarios engage large amounts of data collection for Internet services.
MRS Spark	Spark is an open-source parallel data processing framework. It helps users easily develop unified big data applications and perform cooperative processing, stream processing, and interactive analysis on data. Spark provides a framework featuring fast calculation, write, and interactive query. Spark has obvious advantages over Hadoop in terms of performance. Spark provides the Spark SQL language similar to SQL statements to process structured data.
MRS ClickHouse	ClickHouse is an open-source columnar database oriented to online analysis and processing. It is independent of the Hadoop big data system and features ultimate compression rate and fast query performance. In addition, ClickHouse supports SQL query and provides good query performance, especially the aggregation analysis and query performance based on large and wide tables. The query speed is one order of magnitude faster than that of other analytical databases. ClickHouse is widely used in various fields such as Internet advertising, apps, web, telecommunications, finance, and IoT. It suits business intelligence ideally.

Data Source Type	Description
MRS Impala	Impala provides fast, interactive SQL queries directly on your Apache Hadoop data stored in HDFS, HBase, or the Object Storage Service (OBS). In addition to using the same unified storage platform, Impala also uses the same metadata, SQL syntax (Hive SQL), ODBC driver, and user interface (Impala query UI in Hue) as Apache Hive. This provides a familiar and unified platform for real-time or batch-oriented queries. Impala is an addition to tools available for querying big data. Impala does not replace the batch processing frameworks built on MapReduce such as Hive. Hive and other frameworks built on MapReduce are best suited for long running batch jobs.
MRS Ranger	Ranger offers a centralized security management framework and supports unified authorization and auditing. It manages fine-grained access control over Hadoop and related components, such as HDFS, Hive, HBase, Kafka, and Storm. You can use the frontend web UI console provided by Ranger to configure policies to control users' access to these components.
MRS Hudi	Hudi is a data lake table format that provides the ability to update and delete data as well as consume new data on HDFS. It supports multiple compute engines and provides insert, update, and delete (IUD) interfaces and streaming primitives, including upsert and incremental pull, over datasets on HDFS. Hudi metadata is stored in Hive, and operations are performed using Spark.
MRS Presto	Presto is an open-source SQL query engine for running interactive analytic queries against data sources of all sizes. It applies to massive structured/semi-structured data analysis, massive multi-dimensional data aggregation/report, ETL, ad-hoc queries, and more scenarios. Presto allows querying data where it lives, including HDFS, Hive, HBase, Cassandra, relational databases, or even proprietary data stores. A Presto query can combine different data sources to perform data analysis across the data sources.

Data Source Type	Description
MRS Doris	Doris is a high-performance, real-time analytical database. It can return query results of mass data in sub-seconds and can support high-concurrency point queries and high-throughput complex analysis. Apache Doris can meet requirements in report analysis, instant query, unified data warehouse building, and data lake federated query.
RDS	HUAWEI CLOUD RDS is an online, out-of-the-box relational database service that is based on the cloud computing platform. It is stable, reliable, scalable, and easy to manage. Currently, DataArts Studio supports only MySQL and PostgreSQL databases in RDS.
MySQL	MySQL is one of the most popular open-source databases. It features excellent performance, uses mature and stable architecture, supports popular applications, adapts to multiple fields and industries, and supports various web applications. It is cost-effective and preferred by small- and medium-sized enterprises.
Oracle	Oracle is a group of software that mainly applied to the distributed database. The Oracle database is one of the most popular Client/Server (C/S) and Browser/Server (B/S) databases. It is also the most widely used database management system in the world. As a general database system, the Oracle database provides complete data management functions. As a relational database, it provides complete relational models. As a distributed database, it implements distributed data processing.
DIS	DIS streams are used to schedule jobs between workspaces. If DIS streams are used, messages can be sent to the DIS streams of another account. Otherwise, messages can be sent only to streams in all regions of the current account.
Rest Client	The Rest Client can be used to execute RESTful requests that are authenticated using IAM tokens or usernames and passwords.
Host Connection	You can connect to a specified host during data development and execute shell or Python scripts on the host through script development and job development. If the host connection information changes, you only need to edit it on the **Host Connections** page, but do not need to edit it in scripts or jobs one by one.

3.2 Managing Data Connections

3.2.1 Creating a Data Connection

You can create data connections by configuring data sources. Based on the data connections of the Management Center, DataArts Studio performs data development, governance, services, and operations on the data lake base.

After the data connection between the development environment and production environment is configured, the data connection in the development environment in the script or job during data development is automatically switched to the data connection in the production environment after the process is released.

Constraints

- RDS data connections depend on OBS. If OBS is unavailable in the same region as DataArts Studio, RDS data connections are not supported.
- For host connections, only Linux hosts are supported.
- If changes occur in the connected data lake (for example, the MRS cluster capacity is expanded), you need to edit and save the connection.
- If the data lake authentication information in a data connection changes (for example, the password expires), the data connection becomes invalid. Ensure that the data lake authentication information is permanently valid to prevent any loss caused by connection failures.

Prerequisites

- You have created a data lake to connect, for example, a database or cloud service supported by DataArts Studio.
  - Before creating a DWS data connection, ensure that you have created a cluster in DWS and have the permissions required to view Key Management Service (KMS) keys.
  - Before creating an MRS connection such as an MRS HBase and MRS Hive connection, ensure that you have bought an MRS cluster and selected required components.
- The data lake to connect communicates with the DataArts Studio instance properly.
  - If the data lake is an on-premises database, a public network or a dedicated connection is required. Ensure that the host where the data source is located can access the public network and the port has been enabled in the firewall rule.
  - If the data lake is a cloud service (such as DWS and MRS), the following requirements must be met for network interconnection:
    - If the CDM cluster in the DataArts Studio instance and the cloud service are in different regions, a public network or a dedicated connection is required.

    - If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**.
    - The cloud service instance and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.

- If the enterprise mode is used, pay attention to the following points:

  In enterprise mode, the development environment and production environment need to be distinguished. Therefore, you need to prepare two sets of data lake services for the production environment and development environment to isolate the development environment from the production environment.

  - **If two clusters are used** for clustered data sources, such as MRS, GaussDB(DWS), RDS, MySQL, Oracle, DIS, and ECS, you can create data connections in Management Center to distinguish data lake services in the development environment from those in the production environment. The data lake is automatically switched during development and production. Therefore, you need to prepare two sets of data lake services. The versions, specifications, components, regions, VPCs, subnets, and related configurations of the two sets of data lake services must be the same. For details on how to create data connections, see **Creating a Data Connection**.
  - For serverless services (such as DLI), DataArts Studio configures the mapping between data lake services in the production environment and development environment through environment isolation in the management center. The corresponding data lake is automatically switched during the development and production processes. Therefore, you need to prepare two sets of queues and database resources in the serverless data lake service and distinguish them by name suffix. For details, see **Configuring Environment Isolation for a Workspace in Enterprise Mode**.
  - For GaussDB(DWS), MRS Hive, and MRS Spark, if you **select the same cluster** when creating a data connection, you must configure database mapping on the **Configure Data Source Resource Mapping** page to isolate the development and production environments. For details, see **DB configuration**.

  For example, if your data lake service is an MRS cluster, you need to prepare two MRS clusters with the same version, specifications, components, region, VPC, and subnet. If some configurations of an MRS cluster are modified, you also need to synchronize the modifications to the other MRS cluster.

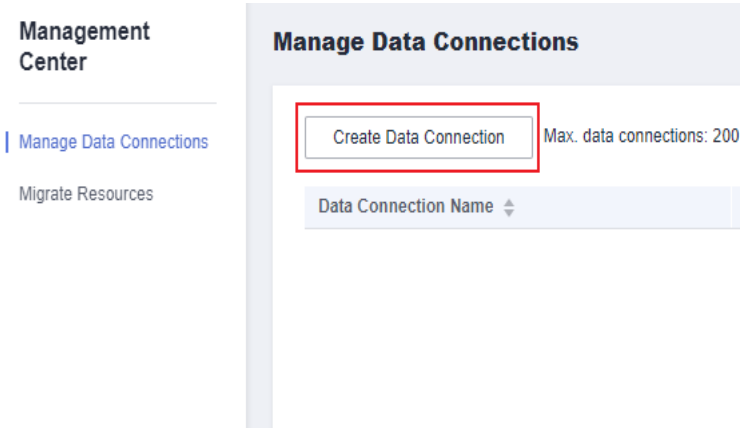
Creating a Data Connection

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

Step 2 On the DataArts Studio console, locate a workspace and click **Management Center**.

Step 3 On the displayed **Manage Data Connections** page, click **Create Data Connection**.

Figure 3-1 Creating a data connection



Step 4 On the displayed page, select a data connection type and configure the parameters listed in **Table 3-3**.

NOTE

- **If two clusters are used** for clustered data sources, such as MRS, GaussDB(DWS), RDS, MySQL, Oracle, DIS, and ECS, you can create data connections in Management Center to distinguish data lake services in the development environment from those in the production environment. The data lake is automatically switched during development and production. Therefore, you need to prepare two sets of data lake services. The versions, specifications, components, regions, VPCs, subnets, and related configurations of the two sets of data lake services must be the same. For details on how to create data connections, see **Creating a Data Connection**.
- For serverless services (such as DLI), DataArts Studio configures the mapping between data lake services in the production environment and development environment through environment isolation in the management center. The corresponding data lake is automatically switched during the development and production processes. Therefore, you need to prepare two sets of queues and database resources in the serverless data lake service and distinguish them by name suffix. For details, see **Configuring Environment Isolation for a Workspace in Enterprise Mode**.
- For GaussDB(DWS), MRS Hive, and MRS Spark, if you **select the same cluster** when creating a data connection, you must configure database mapping on the **Configure Data Source Resource Mapping** page to isolate the development and production environments. For details, see **DB configuration**.

Table 3-3 Data connection parameters

Data Connection Type	Description
DWS	See **Configuring a DWS Connection**.
DLI	See **Configuring a DLI Connection**.
MRS Hive	See **Configuring an MRS Hive Connection**.

Data Connection Type	Description
MRS HBase	See **Configuring an MRS HBase Connection**.
MRS Kafka	See **Configuring an MRS Kafka Connection**.
MRS Spark	See **Configuring an MRS Spark Connection**.
MRS ClickHouse	See **Configuring an MRS ClickHouse Connection**.
MRS Hetu	See **Configuring an MRS Hetu Connection**.
MRS Impala	See **Configuring an MRS Impala Connection**.
MRS Presto	See **Configuring an MRS Presto Connection**.
MRS Doris	See **Configuring an MRS Doris Connection**.
RDS	See **Configuring an RDS Connection**. The RDS connection can connect to relational databases such as RDS for MySQL, PostgreSQL, DM, SQL Server, and SAP HANA.
MySQL (pending offline)	You are not advised to select this connection type. Instead, You are advised to select **RDS**. For details, see **Configuring an RDS Connection**.
Oracle	See **Configuring an Oracle Connection**.
DIS	See **Configuring a DIS Connection**.
Host Connection	See **Configuring a Host Connection**.

Step 5 Click **Test** to test connectivity of the data connection. If the test fails, the data connection cannot be created.

Step 6 After the test is successful, click **Save**. The system will create the data connection for you.

----End

Related Operations

- Edit a data connection: In the data connection list, locate a connection and click **Edit** in the **Operation** column. On the displayed page, modify the parameters listed in **Table 3-3** as needed.

  NOTE

  If you do not want to change the password, you do not need to set it. The system automatically uses the password set when the connection was created.

  Click **Test** to check whether the data connection is normal. If the connection is normal, click **Save**. If the connection is abnormal, the data connection cannot be created. Modify the connection parameters as prompted and try again.

- Delete a data connection: In the data connection list, locate a connection and click **Delete** in the **Operation** column. In the displayed dialog box, confirm the data connection information, and click **OK**.

  If the connection to be deleted is being used, it cannot be deleted directly. In this case, you need to stop the connection from being used on the console of each component and try again.

  NOTE

  If a data connection is deleted, the data table information of the data connection will also be deleted. Exercise caution when performing this operation.

3.2.2 Configuring a DWS Connection

Table 3-4 DWS connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**DWS** is selected and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
SSL Encryption	Yes	DWS supports SSL encryption and certificate authentication for communication between the client and server. You can use **SSL Connection** to set the communication mode. If **SSL Connection** is enabled, only SSL encryption can be used. If **SSL Connection** is disabled, both modes can be used. This function is disabled by default.

Parameter	Mandatory	Description
Manual	Yes	Select either of the following modes: • **Cluster Name Mode**: Select an existing cluster. • **Connection String Mode**: Enter the IP address/ domain name and port of the corresponding cluster and enable the communication between the connection's agent (CDM cluster) and the DWS cluster.
DWS Cluster Name	Yes	This parameter is mandatory when **Manual** is set to **Cluster Name Mode**. Select a DWS cluster from all the DWS clusters with the same project ID and enterprise project.
IP Address or Domain Name	Yes	This parameter is mandatory when **Manual** is set to **Connection String Mode**. This parameter indicates the address for accessing the cluster database through an internal network. Enter an IP address or domain name. The IP address or domain name is automatically generated during cluster creation. You can obtain them on the management console by performing the following operations: 1. Log in to the GaussDB(DWS) console. 2. In the left navigation pane, choose **Instances**. 3. Click the name of an instance to enter the basic information page. In the **Connection Information** area, you can obtain the private IP address, domain name, and port number.
Port	Yes	This parameter is mandatory when **Manual** is set to **Connection String Mode**. This parameter indicates the database port number specified during the DWS cluster creation. Ensure that you have enabled this port in the security group rule so that the DataArts Studio instance can connect to the database in the DWS cluster through this port.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source

Parameter	Mandatory	Description
Agent	Yes	Data Warehouse Service (DWS) is not a fully managed service and thus cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating a DWS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with the DWS cluster. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as the DWS cluster. The security group rule must also allow the CDM cluster communicate with the DWS cluster. **NOTE** If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
Data Source Authentication and Other Function Configuration		
Username	Yes	The database username, which is specified when the DWS cluster is created.
Password	Yes	The password for accessing the database, which is specified when the DWS cluster is created.

3.2.3 Configuring a DLI Connection

Table 3-5 DLI connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**DLI** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).

Parameter	Mandatory	Description
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.

3.2.4 Configuring an MRS Hive Connection

Table 3-6 MRS Hive connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**MRS Hive** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		

Parameter	Mandatory	Description
Connection Type	Yes	Connection type. **Proxy connection** is recommended. • **Proxy connection**: An agent (CDM cluster) is used to access MRS clusters. This method supports all versions of MRS clusters. • **MRS API connection**: MRS APIs are used to access MRS clusters. This method supports only MRS clusters of the 2.*X* or a later version. When you select **MRS API connection**, pay attention to the following restrictions: 1. The MRS API connection is available only for DataArts Factory. 2. In DataArts Factory, you cannot view or manage the databases, data tables, and fields of the connection in a visualized manner. If an MRS cluster of version 3.2.1 or later is connected, you can view rather than manage the databases, data tables, and fields of the connection in a visualized manner. 3. When the SQL editor of DataArts Factory is used to run SQL statements, the execution results can be displayed only in logs. **NOTE** Select **Proxy connection** for **Connection Type** so that the DataArts Architecture, DataArts Quality, DataArts Catalog, and DataArts DataService components can use the MRS connection.
Manual	Yes	This parameter is mandatory when **Connection Type** is set to **Proxy connection**. Select the connection mode. If you do not need to access MRS clusters in other projects or enterprise projects, select **Cluster Name Mode**. • **Cluster Name Mode**: Select an existing cluster. You can only connect to an MRS cluster in the same project and enterprise project. • If you select **Connection String Mode**, you can set **Manager IP** and enable communication between this connection's agent (CDM cluster) and an MRS cluster in another project or enterprise project so that you can access the MRS cluster.

Parameter	Mandatory	Description
Manager IP	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. Set this parameter to the floating IP address of MRS Manager. Only MRS clusters are supported. A Hadoop cluster can be connected only after it is managed by MRS. You can click **Select** next to the text box and select an MRS cluster in the same project and enterprise project. If you want to access an MRS cluster in another project or enterprise project, obtain and enter the floating IP address of MRS Manager and ensure that the connection's agent (CDM cluster) can communicate with the tenant-plane MRS cluster. To obtain the floating IP address of MRS Manager, log in to the active master node of the MRS cluster and run the **ifconfig** command. In the command output, the IP address of **eth0:wsom** is the floating IP address of MRS Manager. For details about how to log in to the master node of the MRS cluster, see **Logging In to an ECS**. Enter multiple IP addresses based on the scenario in sequence and separate them with commas (,), for example, **127.0.0.1** or **127.0.0.1,127.0.0.2,127.0.0.3**. • If you enter one IP address, enter the management-plane floating IP address of the MRS cluster. • If you enter three IP addresses, enter the IP address of the active node on the MRS cluster service plane, IP address of the standby node on the MRS cluster service plane, and the floating IP address of the MRS cluster management plane.

Parameter	Mandatory	Description
MRS Cluster Name	Yes	This parameter is mandatory when **MRS API connection** is selected for **Connection Type** or **Cluster Name Mode** is selected for **Manual**. The name of the MRS cluster. Select an MRS cluster that Hive belongs to. Only MRS clusters are supported. A Hadoop cluster can be selected only after it is managed by MRS. All the MRS clusters with the same project ID and enterprise project are displayed. If the connection fails after you select a cluster, check whether the MRS cluster can communicate with the CDM instance which functions as the agent. They can communicate with each other in the following scenarios: • If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule. • If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**. • The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.
KMS Key	No	This parameter is mandatory when **Connection Type** is set to **Proxy connection**. KMS key used to encrypt and decrypt the authentication information for the data source

Parameter	Mandatory	Description
Agent	Yes	This parameter is mandatory when **Connection Type** is set to **Proxy connection**. MRS is not a fully managed service and cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating an MRS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with the MRS cluster. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as the MRS cluster. The security group rule must also allow the CDM cluster communicate with the MRS cluster. **NOTE** • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster cannot connect to multiple MRS security clusters. You are advised to plan multiple agents which are mapped to MRS security clusters one by one. • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
Data Source Authentication and Other Function Configuration		
Authentication Method	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. It specifies the authentication method used for accessing the MRS cluster. The following options are available: • **SIMPLE**: for non-security mode • **KERBEROS**: for security mode

Parameter	Mandatory	Description
Username	Yes	Human-machine user of the MRS cluster. This parameter is mandatory when **Connection Type** is set to **Proxy connection**. If a new MRS user is used for connection, you need to log in to Manager and change the initial password. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user whose password never expires by referring to **Creating a Kerberos Authentication User for an MRS Security Cluster**. When creating an MRS data connection, set **Username** and **Password** to the new MRS username and password. **NOTE** • For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components. • For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections. • You are advised to set a user password that never expires to prevent connection failures and service loss caused by password expiration.
Password	Yes	The password for accessing the MRS cluster. This parameter is mandatory when **Connection Type** is set to **Proxy connection**.
Enable ldap	No	This parameter is available when **Connection Type** is set to **Proxy connection**. If LDAP authentication is enabled for an external LDAP server connected to MRS Hive, the LDAP username and password are required for authenticating the connection to MRS Hive. In this case, this option must be enabled. Otherwise, the connection will fail.
ldapUsername	Yes	This parameter is mandatory when **Enable ldap** is enabled. Enter the username configured when LDAP authentication was enabled for MRS Hive.

Parameter	Mandatory	Description
ldapPassword	Yes	This parameter is mandatory when **Enable ldap** is enabled. Enter the password configured when LDAP authentication was enabled for MRS Hive.

Creating a Kerberos Authentication User for an MRS Security Cluster

To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. To create an MRS user, perform the following steps:

For clusters of MRS 3.x:

1. Log in to MRS Manager as user **admin**.
2. Choose **System** > **Permission** > **Security Policy** > **Password Policy**. Click **Add Password Policy** and add a policy under which the password never expires.
   - Set **Password Policy Name** to **neverexp**.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.
   - Retain the default values for other parameters.
3. Choose **System** > **Permission** > **User**. On the page displayed, click **Create** to add a dedicated user as the Kerberos authentication user and set the password policy to **neverexp**. Select the user group **superGroup** for the user, and assign all roles to the user.

   NOTE

   - For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components.
   - For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
   - A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.
4. Log in to Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

NOTE

- When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
- When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

For clusters of MRS 2.x or earlier:

1. Log in to the MRS Manager as user **admin**.
2. On FusionInsight Manager, choose **System Settings** and click **Configure Password Policy** to modify the password policy.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.
   - Retain the default values for other parameters.
3. Choose **System** > **Manage User**. On the page displayed, add a dedicated user as the Kerberos authentication user. Select the user group **superGroup** for the user, and assign all roles to the user.

   NOTE

   - For clusters of MRS 2.*x* or earlier, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
   - A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.
4. Log in to MRS Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

NOTE

- When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
- When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

3.2.5 Configuring an MRS HBase Connection

Table 3-7 MRS HBase connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**MRS HBase** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		

Parameter	Mandatory	Description
Manual	Yes	Select the connection mode. If you do not need to access MRS clusters in other projects or enterprise projects, select **Cluster Name Mode**. • **Cluster Name Mode**: Select an existing cluster. You can only connect to an MRS cluster in the same project and enterprise project. • If you select **Connection String Mode**, you can set **Manager IP** and enable communication between this connection's agent (CDM cluster) and an MRS cluster in another project or enterprise project so that you can access the MRS cluster.
Manager IP	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. Set this parameter to the floating IP address of MRS Manager. Only MRS clusters are supported. A Hadoop cluster can be connected only after it is managed by MRS. You can click **Select** next to the text box and select an MRS cluster in the same project and enterprise project. If you want to access an MRS cluster in another project or enterprise project, obtain and enter the floating IP address of MRS Manager and ensure that the connection's agent (CDM cluster) can communicate with the tenant-plane MRS cluster. To obtain the floating IP address of MRS Manager, log in to the active master node of the MRS cluster and run the **ifconfig** command. In the command output, the IP address of **eth0:wsom** is the floating IP address of MRS Manager. For details about how to log in to the master node of the MRS cluster, see **Logging In to an ECS**. Enter multiple IP addresses based on the scenario in sequence and separate them with commas (,), for example, **127.0.0.1** or **127.0.0.1,127.0.0.2,127.0.0.3**. • If you enter one IP address, enter the management-plane floating IP address of the MRS cluster. • If you enter three IP addresses, enter the IP address of the active node on the MRS cluster service plane, IP address of the standby node on the MRS cluster service plane, and the floating IP address of the MRS cluster management plane.

Parameter	Mandatory	Description
MRS Cluster Name	Yes	This parameter is mandatory when **Cluster Name Mode** is selected for **Manual**. The name of the MRS cluster. Select an MRS cluster that Hive belongs to. Only MRS clusters are supported. A Hadoop cluster can be selected only after it is managed by MRS. All the MRS clusters with the same project ID and enterprise project are displayed. If the connection fails after you select a cluster, check whether the MRS cluster can communicate with the CDM instance which functions as the agent. They can communicate with each other in the following scenarios: • If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule. • If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**. • The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source

Parameter	Mandatory	Description
Agent	Yes	MRS is not a fully managed service and cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating an MRS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with the MRS cluster. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as the MRS cluster. The security group rule must also allow the CDM cluster communicate with the MRS cluster. **NOTE** • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster cannot connect to multiple MRS security clusters. You are advised to plan multiple agents which are mapped to MRS security clusters one by one. • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
Data Source Authentication and Other Function Configuration		
Authentication Method	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. It specifies the authentication method used for accessing the MRS cluster. The following options are available: • **SIMPLE**: for non-security mode • **KERBEROS**: for security mode

Parameter	Mandatory	Description
Username	Yes	Username of the MRS cluster If a new MRS user is used for connection, you need to log in to Manager and change the initial password. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user whose password never expires by referring to **Creating a Kerberos Authentication User for an MRS Security Cluster**. When creating an MRS data connection, set **Username** and **Password** to the new MRS username and password. **NOTE** • For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components. • For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections. • You are advised to set a user password that never expires to prevent connection failures and service loss caused by password expiration.
Password	Yes	Password for accessing the MRS cluster.

Creating a Kerberos Authentication User for an MRS Security Cluster

To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. To create an MRS user, perform the following steps:

For clusters of MRS 3.x:

1. Log in to MRS Manager as user **admin**.
2. Choose **System** > **Permission** > **Security Policy** > **Password Policy**. Click **Add Password Policy** and add a policy under which the password never expires.
   - Set **Password Policy Name** to **neverexp**.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.

- Retain the default values for other parameters.

3. Choose **System** > **Permission** > **User**. On the page displayed, click **Create** to add a dedicated user as the Kerberos authentication user and set the password policy to **neverexp**. Select the user group **superGroup** for the user, and assign all roles to the user.

   NOTE

   - For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components.
   - For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
   - A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.

4. Log in to Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.

   a. Log in to the MRS console.

   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.

   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

      NOTE

      - When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
      - When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

For clusters of MRS 2.x or earlier:

1. Log in to the MRS Manager as user **admin**.
2. On FusionInsight Manager, choose **System Settings** and click **Configure Password Policy** to modify the password policy.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.
   - Retain the default values for other parameters.
3. Choose **System** > **Manage User**. On the page displayed, add a dedicated user as the Kerberos authentication user. Select the user group **superGroup** for the user, and assign all roles to the user.

NOTE

- For clusters of MRS 2.*x* or earlier, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
- A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.

4. Log in to MRS Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

      NOTE

      - When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
      - When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

3.2.6 Configuring an MRS Kafka Connection

Table 3-8 MRS Kafka connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**MRS Kafka** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.

Parameter	Mandatory	Description
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
Manual	Yes	Select the connection mode. If you do not need to access MRS clusters in other projects or enterprise projects, select **Cluster Name Mode**. • **Cluster Name Mode**: Select an existing cluster. You can only connect to an MRS cluster in the same project and enterprise project. • If you select **Connection String Mode**, you can set **Manager IP** and enable communication between this connection's agent (CDM cluster) and an MRS cluster in another project or enterprise project so that you can access the MRS cluster.

Parameter	Mandatory	Description
Manager IP	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. Set this parameter to the floating IP address of MRS Manager. Only MRS clusters are supported. A Hadoop cluster can be connected only after it is managed by MRS. You can click **Select** next to the text box and select an MRS cluster in the same project and enterprise project. If you want to access an MRS cluster in another project or enterprise project, obtain and enter the floating IP address of MRS Manager and ensure that the connection's agent (CDM cluster) can communicate with the tenant-plane MRS cluster. To obtain the floating IP address of MRS Manager, log in to the active master node of the MRS cluster and run the **ifconfig** command. In the command output, the IP address of **eth0:wsom** is the floating IP address of MRS Manager. For details about how to log in to the master node of the MRS cluster, see **Logging In to an ECS**. Enter multiple IP addresses based on the scenario in sequence and separate them with commas (,), for example, **127.0.0.1** or **127.0.0.1,127.0.0.2,127.0.0.3**. • If you enter one IP address, enter the management-plane floating IP address of the MRS cluster. • If you enter three IP addresses, enter the IP address of the active node on the MRS cluster service plane, IP address of the standby node on the MRS cluster service plane, and the floating IP address of the MRS cluster management plane.

Parameter	Mandatory	Description
MRS Cluster Name	Yes	This parameter is mandatory when **Cluster Name Mode** is selected for **Manual**. The name of the MRS cluster. Select an MRS cluster that Hive belongs to. Only MRS clusters are supported. A Hadoop cluster can be selected only after it is managed by MRS. All the MRS clusters with the same project ID and enterprise project are displayed. If the connection fails after you select a cluster, check whether the MRS cluster can communicate with the CDM instance which functions as the agent. They can communicate with each other in the following scenarios: • If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule. • If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**. • The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source

Parameter	Mandatory	Description
Agent	Yes	MRS is not a fully managed service and cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating an MRS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with the MRS cluster. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as the MRS cluster. The security group rule must also allow the CDM cluster communicate with the MRS cluster. **NOTE** ● If a CDM cluster functions as the agent for a data connection in Management Center, the cluster cannot connect to multiple MRS security clusters. You are advised to plan multiple agents which are mapped to MRS security clusters one by one. ● If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
Data Source Authentication and Other Function Configuration		
Authentication Method	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. It specifies the authentication method used for accessing the MRS cluster. The following options are available: ● **SIMPLE**: for non-security mode ● **KERBEROS**: for security mode

Parameter	Mandatory	Description
Username	Yes	Username of the MRS cluster If a new MRS user is used for connection, you need to log in to Manager and change the initial password. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user whose password never expires by referring to **Creating a Kerberos Authentication User for an MRS Security Cluster**. When creating an MRS data connection, set **Username** and **Password** to the new MRS username and password. **NOTE** • For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components. • For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections. • You are advised to set a user password that never expires to prevent connection failures and service loss caused by password expiration.
Password	Yes	Password for accessing the MRS cluster.

Creating a Kerberos Authentication User for an MRS Security Cluster

To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. To create an MRS user, perform the following steps:

For clusters of MRS 3.x:

1. Log in to MRS Manager as user **admin**.
2. Choose **System** > **Permission** > **Security Policy** > **Password Policy**. Click **Add Password Policy** and add a policy under which the password never expires.
   - Set **Password Policy Name** to **neverexp**.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.

- Retain the default values for other parameters.

3. Choose **System** > **Permission** > **User**. On the page displayed, click **Create** to add a dedicated user as the Kerberos authentication user and set the password policy to **neverexp**. Select the user group **superGroup** for the user, and assign all roles to the user.

   NOTE

   - For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components.
   - For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
   - A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.

4. Log in to Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

      NOTE

      - When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
      - When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

For clusters of MRS 2.x or earlier:

1. Log in to the MRS Manager as user **admin**.
2. On FusionInsight Manager, choose **System Settings** and click **Configure Password Policy** to modify the password policy.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.
   - Retain the default values for other parameters.
3. Choose **System** > **Manage User**. On the page displayed, add a dedicated user as the Kerberos authentication user. Select the user group **superGroup** for the user, and assign all roles to the user.

NOTE

- For clusters of MRS 2.*x* or earlier, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
- A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.

4. Log in to MRS Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

      NOTE

      - When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
      - When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

3.2.7 Configuring an MRS Spark Connection

Table 3-9 MRS Spark connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**MRS Spark** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.

Parameter	Mandatory	Description
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
Connection Type	Yes	Connection type. **Proxy connection** is recommended. • **Proxy connection**: An agent (CDM cluster) is used to access MRS clusters. This method supports all versions of MRS clusters. • **MRS API connection**: MRS APIs are used to access MRS clusters. This method supports only MRS clusters of the 2.*X* or a later version. When you select **MRS API connection**, pay attention to the following restrictions: 1. The MRS API connection is available only for DataArts Factory. 2. In DataArts Factory, you cannot view or manage the databases, data tables, and fields of the connection in a visualized manner. If an MRS cluster of version 3.2.1 or later is connected, you can view rather than manage the databases, data tables, and fields of the connection in a visualized manner. 3. When the SQL editor of DataArts Factory is used to run SQL statements, the execution results can be displayed only in logs. **NOTE** MRS Spark data connections in MRS API mode apply to data development, while MRS Spark data connections in proxy mode apply to data governance. • To ensure that required resources (such as threads, memory, CPUs, and MRS resource queues) can be independently configured for each Spark SQL job in data development scenarios, select **MRS API connection**. If you select **Proxy connection**, resources cannot be configured independently for each Spark SQL job. • To ensure that other components such as DataArts Architecture can use this connection, select **Proxy connection**.

Parameter	Mandatory	Description
Manual	Yes	This parameter is mandatory when **Connection Type** is set to **Proxy connection**. Select the connection mode. If you do not need to access MRS clusters in other projects or enterprise projects, select **Cluster Name Mode**. • **Cluster Name Mode**: Select an existing cluster. You can only connect to an MRS cluster in the same project and enterprise project. • If you select **Connection String Mode**, you can set **Manager IP** and enable communication between this connection's agent (CDM cluster) and an MRS cluster in another project or enterprise project so that you can access the MRS cluster.
Manager IP	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. Set this parameter to the floating IP address of MRS Manager. Only MRS clusters are supported. A Hadoop cluster can be connected only after it is managed by MRS. You can click **Select** next to the text box and select an MRS cluster in the same project and enterprise project. If you want to access an MRS cluster in another project or enterprise project, obtain and enter the floating IP address of MRS Manager and ensure that the connection's agent (CDM cluster) can communicate with the tenant-plane MRS cluster. To obtain the floating IP address of MRS Manager, log in to the active master node of the MRS cluster and run the **ifconfig** command. In the command output, the IP address of **eth0:wsom** is the floating IP address of MRS Manager. For details about how to log in to the master node of the MRS cluster, see **Logging In to an ECS**. Enter multiple IP addresses based on the scenario in sequence and separate them with commas (,), for example, **127.0.0.1** or **127.0.0.1,127.0.0.2,127.0.0.3**. • If you enter one IP address, enter the management-plane floating IP address of the MRS cluster. • If you enter three IP addresses, enter the IP address of the active node on the MRS cluster service plane, IP address of the standby node on the MRS cluster service plane, and the floating IP address of the MRS cluster management plane.

Parameter	Mandatory	Description
MRS Cluster Name	Yes	This parameter is mandatory when **MRS API connection** is selected for **Connection Type** or **Cluster Name Mode** is selected for **Manual**. The name of the MRS cluster. Select an MRS cluster that Hive belongs to. Only MRS clusters are supported. A Hadoop cluster can be selected only after it is managed by MRS. All the MRS clusters with the same project ID and enterprise project are displayed. If the connection fails after you select a cluster, check whether the MRS cluster can communicate with the CDM instance which functions as the agent. They can communicate with each other in the following scenarios: • If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule. • If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**. • The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.
KMS Key	Yes	This parameter is mandatory when **Connection Type** is set to **Proxy connection**. KMS key used to encrypt and decrypt the authentication information for the data source

Parameter	Mandatory	Description
Agent	Yes	This parameter is mandatory when **Connection Type** is set to **Proxy connection**. MRS is not a fully managed service and cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating an MRS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with the MRS cluster. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as the MRS cluster. The security group rule must also allow the CDM cluster communicate with the MRS cluster. **NOTE** • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster cannot connect to multiple MRS security clusters. You are advised to plan multiple agents which are mapped to MRS security clusters one by one. • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
Data Source Authentication and Other Function Configuration		
Authentication Method	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. It specifies the authentication method used for accessing the MRS cluster. The following options are available: • **SIMPLE**: for non-security mode • **KERBEROS**: for security mode
MRS Version	No	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. Select the MRS cluster version.
Component Name	No	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. Select the Spark version.

Parameter	Mandatory	Description
Username	Yes	Human-machine user of the MRS cluster. This parameter is mandatory when **Connection Type** is set to **Proxy connection**. If a new MRS user is used for connection, you need to log in to Manager and change the initial password. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user whose password never expires by referring to **Creating a Kerberos Authentication User for an MRS Security Cluster**. When creating an MRS data connection, set **Username** and **Password** to the new MRS username and password. **NOTE** • For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components. • For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections. • You are advised to set a user password that never expires to prevent connection failures and service loss caused by password expiration.
Password	Yes	The password for accessing the MRS cluster. This parameter is mandatory when **Connection Type** is set to **Proxy connection**.

Creating a Kerberos Authentication User for an MRS Security Cluster

To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. To create an MRS user, perform the following steps:

For clusters of MRS 3.x:

1. Log in to MRS Manager as user **admin**.
2. Choose **System** > **Permission** > **Security Policy** > **Password Policy**. Click **Add Password Policy** and add a policy under which the password never expires.

- Set **Password Policy Name** to **neverexp**.
- Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
- Set **Password Expiration Notification (Days)** to **0**.
- Retain the default values for other parameters.

3. Choose **System** > **Permission** > **User**. On the page displayed, click **Create** to add a dedicated user as the Kerberos authentication user and set the password policy to **neverexp**. Select the user group **superGroup** for the user, and assign all roles to the user.

   NOTE

   - For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components.
   - For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
   - A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.

4. Log in to Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

      NOTE

      - When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
      - When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

For clusters of MRS 2.x or earlier:

1. Log in to the MRS Manager as user **admin**.
2. On FusionInsight Manager, choose **System Settings** and click **Configure Password Policy** to modify the password policy.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.

– Retain the default values for other parameters.

3. Choose **System** > **Manage User**. On the page displayed, add a dedicated user as the Kerberos authentication user. Select the user group **superGroup** for the user, and assign all roles to the user.

   NOTE

   - For clusters of MRS 2.*x* or earlier, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
   - A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.

4. Log in to MRS Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

      NOTE

      - When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
      - When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

3.2.8 Configuring an MRS ClickHouse Connection

Table 3-10 MRS ClickHouse connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**MRS ClickHouse** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).

Parameter	Mandatory	Description
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
Manual	Yes	Select the connection mode. If you do not need to access MRS clusters in other projects or enterprise projects, select **Cluster Name Mode**. • **Cluster Name Mode**: Select an existing cluster. You can only connect to an MRS cluster in the same project and enterprise project. • If you select **Connection String Mode**, you can set **Manager IP** and enable communication between this connection's agent (CDM cluster) and an MRS cluster in another project or enterprise project so that you can access the MRS cluster.

Parameter	Mandatory	Description
Manager IP	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. Set this parameter to the floating IP address of MRS Manager. Only MRS clusters are supported. A Hadoop cluster can be connected only after it is managed by MRS. You can click **Select** next to the text box and select an MRS cluster in the same project and enterprise project. If you want to access an MRS cluster in another project or enterprise project, obtain and enter the floating IP address of MRS Manager and ensure that the connection's agent (CDM cluster) can communicate with the tenant-plane MRS cluster. To obtain the floating IP address of MRS Manager, log in to the active master node of the MRS cluster and run the **ifconfig** command. In the command output, the IP address of **eth0:wsom** is the floating IP address of MRS Manager. For details about how to log in to the master node of the MRS cluster, see **Logging In to an ECS**. Enter multiple IP addresses based on the scenario in sequence and separate them with commas (,), for example, **127.0.0.1** or **127.0.0.1,127.0.0.2,127.0.0.3**. • If you enter one IP address, enter the management-plane floating IP address of the MRS cluster. • If you enter three IP addresses, enter the IP address of the active node on the MRS cluster service plane, IP address of the standby node on the MRS cluster service plane, and the floating IP address of the MRS cluster management plane.

Parameter	Mandatory	Description
MRS Cluster Name	Yes	This parameter is mandatory when **Cluster Name Mode** is selected for **Manual**. The name of the MRS cluster. Select an MRS cluster that Hive belongs to. Only MRS clusters are supported. A Hadoop cluster can be selected only after it is managed by MRS. All the MRS clusters with the same project ID and enterprise project are displayed. If the connection fails after you select a cluster, check whether the MRS cluster can communicate with the CDM instance which functions as the agent. They can communicate with each other in the following scenarios: • If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule. • If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**. • The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source

Parameter	Mandatory	Description
Agent	Yes	MRS is not a fully managed service and cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating an MRS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with the MRS cluster. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as the MRS cluster. The security group rule must also allow the CDM cluster communicate with the MRS cluster. **NOTE** • MRS ClickHouse connections are supported only in CDM 2.9.2 and later versions. • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster cannot connect to multiple MRS security clusters. You are advised to plan multiple agents which are mapped to MRS security clusters one by one. • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
Data Source Authentication and Other Function Configuration		
Authentication Method	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. It specifies the authentication method used for accessing the MRS cluster. The following options are available: • **SIMPLE**: for non-security mode • **KERBEROS**: for security mode

Parameter	Mandatory	Description
Username	Yes	Username of the MRS cluster If a new MRS user is used for connection, you need to log in to Manager and change the initial password. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user whose password never expires by referring to **Creating a Kerberos Authentication User for an MRS Security Cluster**. When creating an MRS data connection, set **Username** and **Password** to the new MRS username and password. **NOTE** • For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components. • For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections. • You are advised to set a user password that never expires to prevent connection failures and service loss caused by password expiration.
Password	Yes	Password for accessing the MRS cluster.

Creating a Kerberos Authentication User for an MRS Security Cluster

To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. To create an MRS user, perform the following steps:

For clusters of MRS 3.x:

1. Log in to MRS Manager as user **admin**.
2. Choose **System** > **Permission** > **Security Policy** > **Password Policy**. Click **Add Password Policy** and add a policy under which the password never expires.
   - Set **Password Policy Name** to **neverexp**.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.

- Retain the default values for other parameters.

3. Choose **System** > **Permission** > **User**. On the page displayed, click **Create** to add a dedicated user as the Kerberos authentication user and set the password policy to **neverexp**. Select the user group **superGroup** for the user, and assign all roles to the user.

   NOTE

   - For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components.
   - For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
   - A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.

4. Log in to Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.

   a. Log in to the MRS console.

   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.

   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

      NOTE

      - When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
      - When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

For clusters of MRS 2.x or earlier:

1. Log in to the MRS Manager as user **admin**.
2. On FusionInsight Manager, choose **System Settings** and click **Configure Password Policy** to modify the password policy.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.
   - Retain the default values for other parameters.
3. Choose **System** > **Manage User**. On the page displayed, add a dedicated user as the Kerberos authentication user. Select the user group **superGroup** for the user, and assign all roles to the user.

NOTE

- For clusters of MRS 2.*x* or earlier, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
- A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.

4. Log in to MRS Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

      NOTE

      - When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
      - When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

3.2.9 Configuring an MRS Hetu Connection

Table 3-11 MRS Hetu connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**MRS Hetu** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.

Parameter	Mandatory	Description
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
Manual	Yes	Select the connection mode. If you do not need to access MRS clusters in other projects or enterprise projects, select **Cluster Name Mode**. • **Cluster Name Mode**: Select an existing cluster. You can only connect to an MRS cluster in the same project and enterprise project. • If you select **Connection String Mode**, you can set **Manager IP** and enable communication between this connection's agent (CDM cluster) and an MRS cluster in another project or enterprise project so that you can access the MRS cluster.

Parameter	Mandatory	Description
Manager IP	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. Set this parameter to the floating IP address of MRS Manager. Only MRS clusters are supported. A Hadoop cluster can be connected only after it is managed by MRS. **NOTE** • MRS clusters of version 3.1.1 and later can be connected. • To connect to MRS clusters of version 3.2.1, add parameter **protocol.v1.alternate-header-name** with value **Presto** in the **coordinator.config.properties** and **worker.config.properties** files for the compute instance on the HetuEngine WebUI. You can click **Select** next to the text box and select an MRS cluster in the same project and enterprise project. If you want to access an MRS cluster in another project or enterprise project, obtain and enter the floating IP address of MRS Manager and ensure that the connection's agent (CDM cluster) can communicate with the tenant-plane MRS cluster. To obtain the floating IP address of MRS Manager, log in to the active master node of the MRS cluster and run the **ifconfig** command. In the command output, the IP address of **eth0:wsom** is the floating IP address of MRS Manager. For details about how to log in to the master node of the MRS cluster, see **Logging In to an ECS**. Enter multiple IP addresses based on the scenario in sequence and separate them with commas (,), for example, **127.0.0.1** or **127.0.0.1,127.0.0.2,127.0.0.3**. • If you enter one IP address, enter the management-plane floating IP address of the MRS cluster. • If you enter three IP addresses, enter the IP address of the active node on the MRS cluster service plane, IP address of the standby node on the MRS cluster service plane, and the floating IP address of the MRS cluster management plane.

Parameter	Mandatory	Description
MRS Cluster Name	Yes	This parameter is mandatory when **Cluster Name Mode** is selected for **Manual**. The name of the MRS cluster. Select an MRS cluster that Hive belongs to. Only MRS clusters are supported. A Hadoop cluster can be selected only after it is managed by MRS. All the MRS clusters with the same project ID and enterprise project are displayed. **NOTE** • MRS clusters of version 3.1.1 and later can be connected. • To connect to MRS clusters of version 3.2.1, add parameter **protocol.v1.alternate-header-name** with value **Presto** in the **coordinator.config.properties** and **worker.config.properties** files for the compute instance on the HetuEngine WebUI. If the connection fails after you select a cluster, check whether the MRS cluster can communicate with the CDM instance which functions as the agent. They can communicate with each other in the following scenarios: • If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule. • If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**. • The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source

Parameter	Mandatory	Description
Agent	Yes	MRS is not a fully managed service and cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating an MRS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with the MRS cluster. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as the MRS cluster. The security group rule must also allow the CDM cluster communicate with the MRS cluster. **NOTE** • MRS Hetu connections are supported only in CDM 2.9.2 and later versions. • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster cannot connect to multiple MRS security clusters. You are advised to plan multiple agents which are mapped to MRS security clusters one by one. • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
hsbroker IP Address List	Yes	IP addresses of the hsbroker nodes of the MRS Hetu component. Use commas (,) to separate multiple IP addresses. To obtain the port number, perform the following operations: 1. Log in to MRS FusionInsight Manager. 2. Choose **Cluster** > **Services** > **HetuEngine** > **Role** > **HSBroker** to obtain the service IP addresses of all HSBroker instances.

Parameter	Mandatory	Description
hsbroker Port	Yes	Port number of the hsbroker node of the MRS Hetu component. To obtain the port number, perform the following operations: 1. Log in to MRS FusionInsight Manager. 2. Choose **Cluster** > **Services** > **HetuEngine** > **Configurations** > **All Configurations** and search for **server.port** on the right to obtain the port number of HSBroker.
Data Source Authentication and Other Function Configuration		
Authentication Method	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. It specifies the authentication method used for accessing the MRS cluster. The following options are available: • **SIMPLE**: for non-security mode • **KERBEROS**: for security mode

Parameter	Mandatory	Description
Username	Yes	Username of the MRS cluster. The user must have permissions of HetuEngine. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user whose password never expires by referring to **Creating a Kerberos Authentication User for an MRS Security Cluster**. When creating an MRS data connection, set **Username** and **Password** to the new MRS username and password. **NOTE** • For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components. • For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections. • You are advised to set a user password that never expires to prevent connection failures and service loss caused by password expiration. **NOTICE** After creating the HetuEngine user, you need to complete the configurations in **Using HetuEngine from Scratch**.
Password	Yes	Password for accessing the MRS cluster.

Creating a Kerberos Authentication User for an MRS Security Cluster

To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. To create an MRS user, perform the following steps:

For clusters of MRS 3.x:

1. Log in to MRS Manager as user **admin**.
2. Choose **System** > **Permission** > **Security Policy** > **Password Policy**. Click **Add Password Policy** and add a policy under which the password never expires.
   - Set **Password Policy Name** to **neverexp**.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.

- Set **Password Expiration Notification (Days)** to **0**.
- Retain the default values for other parameters.

3. Choose **System** > **Permission** > **User**. On the page displayed, click **Create** to add a dedicated user as the Kerberos authentication user and set the password policy to **neverexp**. Select the user group **superGroup** for the user, and assign all roles to the user.

   NOTE

   - For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components.
   - For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
   - A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.

4. Log in to Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

      NOTE

      - When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
      - When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

For clusters of MRS 2.x or earlier:

1. Log in to the MRS Manager as user **admin**.
2. On FusionInsight Manager, choose **System Settings** and click **Configure Password Policy** to modify the password policy.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.
   - Retain the default values for other parameters.
3. Choose **System** > **Manage User**. On the page displayed, add a dedicated user as the Kerberos authentication user. Select the user group **superGroup** for the user, and assign all roles to the user.

NOTE

- For clusters of MRS 2.*x* or earlier, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
- A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.

4. Log in to MRS Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

      NOTE

      - When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
      - When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

3.2.10 Configuring an MRS Impala Connection

Table 3-12 MRS Impala connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**MRS Impala** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.

Parameter	Mandatory	Description
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
Manual	Yes	Select the connection mode. If you do not need to access MRS clusters in other projects or enterprise projects, select **Cluster Name Mode**. • **Cluster Name Mode**: Select an existing cluster. You can only connect to an MRS cluster in the same project and enterprise project. • If you select **Connection String Mode**, you can set **Manager IP** and enable communication between this connection's agent (CDM cluster) and an MRS cluster in another project or enterprise project so that you can access the MRS cluster.

Parameter	Mandatory	Description
Manager IP	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. Set this parameter to the floating IP address of MRS Manager. Only MRS clusters are supported. A Hadoop cluster can be connected only after it is managed by MRS. You can click **Select** next to the text box and select an MRS cluster in the same project and enterprise project. If you want to access an MRS cluster in another project or enterprise project, obtain and enter the floating IP address of MRS Manager and ensure that the connection's agent (CDM cluster) can communicate with the tenant-plane MRS cluster. To obtain the floating IP address of MRS Manager, log in to the active master node of the MRS cluster and run the **ifconfig** command. In the command output, the IP address of **eth0:wsom** is the floating IP address of MRS Manager. For details about how to log in to the master node of the MRS cluster, see **Logging In to an ECS**. Enter multiple IP addresses based on the scenario in sequence and separate them with commas (,), for example, **127.0.0.1** or **127.0.0.1,127.0.0.2,127.0.0.3**. • If you enter one IP address, enter the management-plane floating IP address of the MRS cluster. • If you enter three IP addresses, enter the IP address of the active node on the MRS cluster service plane, IP address of the standby node on the MRS cluster service plane, and the floating IP address of the MRS cluster management plane.

Parameter	Mandatory	Description
MRS Cluster Name	Yes	This parameter is mandatory when **Cluster Name Mode** is selected for **Manual**. The name of the MRS cluster. Select an MRS cluster that Hive belongs to. Only MRS clusters are supported. A Hadoop cluster can be selected only after it is managed by MRS. All the MRS clusters with the same project ID and enterprise project are displayed. If the connection fails after you select a cluster, check whether the MRS cluster can communicate with the CDM instance which functions as the agent. They can communicate with each other in the following scenarios: • If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule. • If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**. • The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source

Parameter	Mandatory	Description
Agent	Yes	MRS is not a fully managed service and cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating an MRS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with the MRS cluster. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as the MRS cluster. The security group rule must also allow the CDM cluster communicate with the MRS cluster. **NOTE** • MRS Impala connections are supported only in CDM 2.9.2 and later versions. • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster cannot connect to multiple MRS security clusters. You are advised to plan multiple agents which are mapped to MRS security clusters one by one. • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
impaladlps	Yes	Management IP address of the Impalad role of the MRS Impala component To obtain it, perform the following operations: 1. Log in to MRS FusionInsight Manager. 2. Choose **Cluster** > **Services** > **Impala** > **Instance** to view the Impalad management IP address.
Data Source Authentication and Other Function Configuration		
Authentication Method	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. It specifies the authentication method used for accessing the MRS cluster. The following options are available: • **SIMPLE**: for non-security mode • **KERBEROS**: for security mode

Parameter	Mandatory	Description
Username	Yes	Human-machine user of the MRS cluster. This parameter is mandatory when **Connection Type** is set to **Proxy connection**. If a new MRS user is used for connection, you need to log in to Manager and change the initial password. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user whose password never expires by referring to **Creating a Kerberos Authentication User for an MRS Security Cluster**. When creating an MRS data connection, set **Username** and **Password** to the new MRS username and password. **NOTE** • For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components. • For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections. • You are advised to set a user password that never expires to prevent connection failures and service loss caused by password expiration.
Password	Yes	The password for accessing the MRS cluster. This parameter is mandatory when **Connection Type** is set to **Proxy connection**.
Enable ldap	No	This parameter is available when **Proxy connection** is selected for **Connection Type**. If LDAP authentication is enabled for an external LDAP server connected to MRS Impala, the LDAP username and password are required for authenticating the connection to MRS Impala. In this case, this option must be enabled. Otherwise, the connection will fail.
ldapUsername	Yes	This parameter is mandatory when **Enable ldap** is enabled. Enter the username configured when LDAP authentication was enabled for MRS Impala.

Parameter	Mandatory	Description
ldapPassword	Yes	This parameter is mandatory when **Enable ldap** is enabled. Enter the password configured when LDAP authentication was enabled for MRS Impala.

Creating a Kerberos Authentication User for an MRS Security Cluster

To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. To create an MRS user, perform the following steps:

For clusters of MRS 3.x:

1. Log in to MRS Manager as user **admin**.
2. Choose **System** > **Permission** > **Security Policy** > **Password Policy**. Click **Add Password Policy** and add a policy under which the password never expires.
   - Set **Password Policy Name** to **neverexp**.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.
   - Retain the default values for other parameters.
3. Choose **System** > **Permission** > **User**. On the page displayed, click **Create** to add a dedicated user as the Kerberos authentication user and set the password policy to **neverexp**. Select the user group **superGroup** for the user, and assign all roles to the user.

   NOTE

   - For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components.
   - For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
   - A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.
4. Log in to Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

NOTE

- When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
- When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

For clusters of MRS 2.x or earlier:

1. Log in to the MRS Manager as user **admin**.
2. On FusionInsight Manager, choose **System Settings** and click **Configure Password Policy** to modify the password policy.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.
   - Retain the default values for other parameters.
3. Choose **System** > **Manage User**. On the page displayed, add a dedicated user as the Kerberos authentication user. Select the user group **superGroup** for the user, and assign all roles to the user.

   NOTE

   - For clusters of MRS 2.*x* or earlier, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
   - A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.
4. Log in to MRS Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

NOTE

- When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
- When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

3.2.11 Configuring an MRS Ranger Connection

Table 3-13 MRS Ranger connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**MRS Ranger** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		

Parameter	Mandatory	Description
Manual	Yes	Select the connection mode. If you do not need to access MRS clusters in other projects or enterprise projects, select **Cluster Name Mode**. • **Cluster Name Mode**: Select an existing cluster. You can only connect to an MRS cluster in the same project and enterprise project. • If you select **Connection String Mode**, you can set **Manager IP** and enable communication between this connection's agent (CDM cluster) and an MRS cluster in another project or enterprise project so that you can access the MRS cluster.

Parameter	Mandatory	Description
Manager IP	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. Set this parameter to the floating IP address of MRS Manager. Only MRS clusters are supported. A Hadoop cluster can be connected only after it is managed by MRS. You can click **Select** next to the text box and select an MRS cluster in the same project and enterprise project. If you want to access an MRS cluster in another project or enterprise project, obtain and enter the floating IP address of MRS Manager and ensure that the connection's agent (CDM cluster) can communicate with the tenant-plane MRS cluster. To obtain the floating IP address of MRS Manager, log in to the active master node of the MRS cluster and run the **ifconfig** command. In the command output, the IP address of **eth0:wsom** is the floating IP address of MRS Manager. For details about how to log in to the master node of the MRS cluster, see **Logging In to an ECS**. Enter multiple IP addresses based on the scenario in sequence and separate them with commas (,), for example, **127.0.0.1** or **127.0.0.1,127.0.0.2,127.0.0.3**. ● If you enter one IP address, enter the management-plane floating IP address of the MRS cluster. ● If you enter three IP addresses, enter the IP address of the active node on the MRS cluster service plane, IP address of the standby node on the MRS cluster service plane, and the floating IP address of the MRS cluster management plane. **NOTE** If the version of the CDM cluster selected as the agent is 2.9.3.300 or earlier, you can only create a connection to MRS Ranger in an MRS cluster in security mode. To create a connection to MRS Ranger in an MRS cluster in non-security mode, ensure that the CDM cluster version is 2.10.0.300 or later, or contact customer service or technical support to upgrade the dlg-agent component in the CDM cluster.

Parameter	Mandatory	Description
MRS Cluster Name	Yes	This parameter is mandatory when **Cluster Name Mode** is selected for **Manual**. The name of the MRS cluster. Select an MRS cluster that Hive belongs to. Only MRS clusters are supported. A Hadoop cluster can be selected only after it is managed by MRS. All the MRS clusters with the same project ID and enterprise project are displayed. If the connection fails after you select a cluster, check whether the MRS cluster can communicate with the CDM instance which functions as the agent. They can communicate with each other in the following scenarios: • If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule. • If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**. • The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace. **NOTE** If the version of the CDM cluster selected as the agent is 2.9.3.300 or earlier, you can only create a connection to MRS Ranger in an MRS cluster in security mode. To create a connection to MRS Ranger in an MRS cluster in non-security mode, ensure that the CDM cluster version is 2.10.0.300 or later, or contact customer service or technical support to upgrade the dlg-agent component in the CDM cluster.

Parameter	Mandatory	Description
IP	Yes	Management IP address of the RangerAdmin role of the MRS Ranger component. Separate multiple IP addresses with commas (,). To obtain the port number, perform the following operations: 1. Log in to MRS FusionInsight Manager. 2. Choose **Cluster** > **Services** > **Ranger** > **Instance** to view the management IP address of the RangerAdmin role.
Port	Yes	Port number of the MRS Ranger instance. To obtain the port number, perform the following operations: 1. Log in to MRS FusionInsight Manager. 2. Choose **Cluster** > **Services** > **Ranger** > **Configurations** > **Basic Configurations**. For an MRS cluster in non-security mode, obtain the port corresponding to the **ranger.service.http.port** parameter. For an MRS cluster in security mode, obtain the port corresponding to the **ranger.service.https.port** parameter.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source

Parameter	Mandatory	Description
Agent	Yes	MRS is not a fully managed service and cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating an MRS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with the MRS cluster. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as the MRS cluster. The security group rule must also allow the CDM cluster communicate with the MRS cluster. **NOTE** • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster cannot connect to multiple MRS security clusters. You are advised to plan multiple agents which are mapped to MRS security clusters one by one. • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
Data Source Authentication and Other Function Configuration		
Authentication Method	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. It specifies the authentication method used for accessing the MRS cluster. The following options are available: • **SIMPLE**: for non-security mode • **KERBEROS**: for security mode

Parameter	Mandatory	Description
Username	Yes	Username of the MRS cluster If a new MRS user is used for connection, you need to log in to Manager and change the initial password. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user whose password never expires by referring to **Creating a Kerberos Authentication User for an MRS Security Cluster**. When creating an MRS data connection, set **Username** and **Password** to the new MRS username and password. **NOTE** • For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components. • For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections. • You are advised to set a user password that never expires to prevent connection failures and service loss caused by password expiration.
Password	Yes	Password for accessing the MRS cluster.

Creating a Kerberos Authentication User for an MRS Security Cluster

To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. To create an MRS user, perform the following steps:

For clusters of MRS 3.x:

1. Log in to MRS Manager as user **admin**.
2. Choose **System** > **Permission** > **Security Policy** > **Password Policy**. Click **Add Password Policy** and add a policy under which the password never expires.
   - Set **Password Policy Name** to **neverexp**.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.

- Set **Password Expiration Notification (Days)** to **0**.
- Retain the default values for other parameters.

3. Choose **System** > **Permission** > **User**. On the page displayed, click **Create** to add a dedicated user as the Kerberos authentication user and set the password policy to **neverexp**. Select the user group **superGroup** for the user, and assign all roles to the user.

   NOTE

   - For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components.
   - For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
   - A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.

4. Log in to Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

      NOTE

      - When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
      - When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

For clusters of MRS 2.x or earlier:

1. Log in to the MRS Manager as user **admin**.
2. On FusionInsight Manager, choose **System Settings** and click **Configure Password Policy** to modify the password policy.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.
   - Retain the default values for other parameters.
3. Choose **System** > **Manage User**. On the page displayed, add a dedicated user as the Kerberos authentication user. Select the user group **superGroup** for the user, and assign all roles to the user.

NOTE

- For clusters of MRS 2.*x* or earlier, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
- A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.

4. Log in to MRS Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

      NOTE

      - When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
      - When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

3.2.12 Configuring an MRS Presto Connection

Table 3-14 MRS Presto connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**MRS Presto** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.

Parameter	Mandatory	Description
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
MRS Cluster Name	Yes	The name of the MRS cluster. Select an MRS cluster that Hive belongs to. Only MRS clusters are supported. A Hadoop cluster can be selected only after it is managed by MRS. All the MRS clusters with the same project ID and enterprise project are displayed. If the connection fails after you select a cluster, check whether the MRS cluster can communicate with the CDM instance which functions as the agent. They can communicate with each other in the following scenarios: • If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule. • If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**. • The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.
Data Source Authentication and Other Function Configuration		
Description	No	You can enter the description of the connection.

3.2.13 Configuring an MRS Doris Connection

Table 3-15 MRS Doris connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**MRS Doris** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
Doris Type	Yes	You can select **MRS Doris** or **CloudTable Doris**.

Parameter	Mandatory	Description
MRS Cluster Name	Yes	This parameter is valid when **Doris Type** is set to **MRS Doris**. The name of the MRS cluster. Select an MRS cluster that Hive belongs to. Only MRS clusters are supported. A Hadoop cluster can be selected only after it is managed by MRS. All the MRS clusters with the same project ID and enterprise project are displayed. If the connection fails after you select a cluster, check whether the MRS cluster can communicate with the CDM instance which functions as the agent. They can communicate with each other in the following scenarios: • If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule. • If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**. • The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace. **NOTE** Only non-security MRS clusters of version 3.2.0 and later support this connection.
FE IP	Yes	IP address of the frontend node of the Doris or Cloud component in the MRS cluster. You can enter one or more IP addresses. Separate multiple IP addresses with commas (,). To obtain them, perform the following operations: 1. Log in to MRS FusionInsight Manager. 2. Choose **Cluster** > **Services** > **Doris** > **Instance** to obtain the management IP address of the FE role.

Parameter	Mandatory	Description
Port	Yes	Port used by the Doris FE to query connections through the MySQL protocol To obtain MRS Doris, perform the following steps: 1. Log in to MRS FusionInsight Manager. 2. Choose **Cluster** > **Services** > **Doris** > **Configurations** > **Basic Configurations**, search for **query_port**, and view the port number.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source
Agent	Yes	MRS is not a fully managed service and cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating an MRS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with the MRS cluster. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as the MRS cluster. The security group rule must also allow the CDM cluster communicate with the MRS cluster. **NOTE** ● If a CDM cluster functions as the agent for a data connection in Management Center, the cluster cannot connect to multiple MRS security clusters. You are advised to plan multiple agents which are mapped to MRS security clusters one by one. ● If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
Data Source Driver Configuration		
Driver Name	Yes	Driver name. Currently, the MySQL JDBC driver is supported. The driver name is **com.mysql.jdbc.Driver**.
Driver Source	Yes	Select the source of the driver file.

Parameter	Mandatory	Description
Driver File Path	Yes	Path of the driver file in the OBS bucket. You need to download the .jar driver file from the corresponding official website and upload it to the OBS bucket. MySQL driver: Obtain the driver from **https://downloads.mysql.com/archives/c-j/**. Version 5.1.48 or later is recommended. If the version is earlier than 5.1.48, error "The db user or password invalid" will be reported. **NOTE** To update the driver, you must restart the CDM cluster in DataArts Migration and then edit the data connection to upload the driver.
Driver File	Yes	This parameter is mandatory when **Driver Source** is set to **Local file**. Select a driver version that adapts to the database type.
Data Source Authentication and Other Function Configuration		
Username	Yes	Username of the MRS or CloudTable cluster. If a new MRS user is used for connection, you need to log in to Manager and change the initial password. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user with a permanent password by referring to **Creating a Kerberos Authentication User for an MRS Security Cluster**. When creating an MRS data connection, set **Username** and **Password** to the new MRS username and password. **NOTE** • For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components. • For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections. • You are advised to set a user password that never expires to prevent connection failures and service loss caused by password expiration.
Password	Yes	It can also be the password for accessing the MRS or CloudTable cluster.

Creating a Kerberos Authentication User for an MRS Security Cluster

To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. To create an MRS user, perform the following steps:

For clusters of MRS 3.x:

1. Log in to MRS Manager as user **admin**.
2. Choose **System** > **Permission** > **Security Policy** > **Password Policy**. Click **Add Password Policy** and add a policy under which the password never expires.
   - Set **Password Policy Name** to **neverexp**.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.
   - Retain the default values for other parameters.
3. Choose **System** > **Permission** > **User**. On the page displayed, click **Create** to add a dedicated user as the Kerberos authentication user and set the password policy to **neverexp**. Select the user group **superGroup** for the user, and assign all roles to the user.

   NOTE

   - For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components.
   - For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
   - A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.
4. Log in to Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

NOTE

- When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
- When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

For clusters of MRS 2.x or earlier:

1. Log in to the MRS Manager as user **admin**.
2. On FusionInsight Manager, choose **System Settings** and click **Configure Password Policy** to modify the password policy.
   - Set **Password Validity Period (Days)** to **0**, indicating that the password never expires.
   - Set **Password Expiration Notification (Days)** to **0**.
   - Retain the default values for other parameters.
3. Choose **System** > **Manage User**. On the page displayed, add a dedicated user as the Kerberos authentication user. Select the user group **superGroup** for the user, and assign all roles to the user.

   NOTE

   - For clusters of MRS 2.*x* or earlier, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center.
   - A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.
4. Log in to MRS Manager as the new user and change the initial password. Otherwise, the connection fails to be created.
5. Synchronize IAM users.
   a. Log in to the MRS console.
   b. Choose **Clusters** > **Active Clusters**, select a running cluster, and click its name to go to its details page.
   c. In the **Basic Information** area of the **Dashboard** page, click **Synchronize** on the right side of **IAM User Sync** to synchronize IAM users.

NOTE

- When the policy of the user group to which the IAM user belongs changes from **MRS ReadOnlyAccess** to **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** (System Security Services Daemon) cache of cluster nodes needs time to be updated. Then, submit a job. Otherwise, the job may fail to be submitted.
- When the policy of the user group to which the IAM user belongs changes from **MRS CommonOperations**, **MRS FullAccess**, or **MRS Administrator** to **MRS ReadOnlyAccess**, wait for 5 minutes until the new policy takes effect after the synchronization is complete because the **SSSD** cache of cluster nodes needs time to be updated.

3.2.14 Configuring an RDS Connection

You can create connections to RDS databases including MySQL, PostgreSQL, SQL Server, SAP HANA, and DM databases.

Table 3-16 RDS connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**RDS** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		

Parameter	Mandatory	Description
IP Address or Domain Name	Yes	Address for accessing the relational database data source. The value can be an IP address or a domain name. • If the data source is RDS or GaussDB, you can obtain the address on the management console by performing the following operations: 1. Log in to the management console of the corresponding cloud service using the account you have obtained. 2. In the left navigation pane, choose **Instances**. 3. Click the name of an instance to enter the basic information page. In the **Connection Information** area, you can obtain the private IP address, domain name, and port number. **NOTE** Only the GaussDB data source supports multiple domain names. Use commas (,) to separate multiple domain names. • If the data source is MySQL, PostgreSQL, or DM, you can obtain the access address from the database administrator.
Port	Yes	Port for accessing the relational database. • If the data source is RDS or GaussDB, you can obtain the address on the management console by performing the following operations: 1. Log in to the management console of the corresponding cloud service using the account you have obtained. 2. In the left navigation pane, choose **Instances**. 3. Click the name of an instance to enter the basic information page. In the **Connection Information** area, you can obtain the private IP address, domain name, and port number. **NOTE** Only the GaussDB data source supports multiple domain names. Use commas (,) to separate multiple domain names. • If the data source is MySQL, PostgreSQL, or DM, you can obtain the access address from the database administrator.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source

Parameter	Mandatory	Description
Agent	Yes	RDS is not a fully managed service and cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating an RDS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with RDS. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as RDS. The security group rule must also allow the CDM cluster to communicate with RDS. **NOTE** If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
Data Source Driver Configuration		
Driver Name	Yes	The name of the driver. The following values are available: • **com.mysql.jdbc.Driver**: Select this driver name for RDS for MySQL or MySQL. • **org.postgresql.Driver**: Select this driver name for RDS for PostgreSQL or PostgreSQL. • **com.microsoft.sqlserver.jdbc.SQLServerDriver**: Select this driver name for RDS for SQL Server. • **dm.jdbc.driver.DmDriver**: Select this driver name for the Dameng database. • **com.huawei.opengauss.jdbc.Driver**: Select this driver name for RDS for GaussDB.

Parameter	Mandatory	Description
Driver File Path	Yes	It specifies the OBS path where the driver file is located. You need to download a .jar driver file from the corresponding official website and upload it to OBS. • MySQL driver: Download it from **https://downloads.mysql.com/archives/c-j/**. The 5.1.48 version is recommended. • PostgreSQL driver: Download it from **https://mvnrepository.com/artifact/org.postgresql/postgresql**. The 42.3.4 version is recommended. • SQL Server driver: Download it from **https://learn.microsoft.com/en-us/sql/connect/jdbc/download-microsoft-jdbc-driver-for-sql-server?view=sql-server-ver16**. The 8.4.1 version is recommended. • Dameng database driver: Obtain **DmJdbcDriver18.jar** from the DM installation directory **/dmdbms/drivers/jdbc**. • GaussDB driver: Search for "JDBC Package, Driver Class, and Environment Class" in *GaussDB User Guide* in the **GaussDB Documentation**, select the document corresponding to the instance version, and obtain the driver package by referring to the document. **NOTE** • The OBS path of the driver file cannot contain Chinese characters. • To update the driver, you must restart the CDM cluster in DataArts Migration and then edit the data connection to upload the driver.
Data Source Authentication and Other Function Configuration		
Username	Yes	The username of the database. The username is required for creating a cluster.
Password	Yes	The password for accessing the database. The password is required for creating a cluster.

3.2.15 Configuring an Oracle Connection

Table 3-17 Oracle connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**ORACLE** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
IP Address or Domain Name	Yes	Address for accessing the database to be connected. You can enter a public/private IP address or a domain name.
Port	Yes	The port of the database to connect.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source

Parameter	Mandatory	Description
Agent	Yes	Oracle is not a fully managed service and cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating an Oracle data connection. If no CDM cluster is available, create one through the DataArts Migration incremental package. As a network proxy, the CDM cluster must be able to communicate with Oracle. **NOTE** If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
Data Source Authentication and Other Function Configuration		
Username	Yes	Username used for accessing the database This account must have the permissions required to read and write data tables and metadata. **NOTE** If you have the CONNECT permission (read-only permission) and are trying to create a connection, a message is displayed indicating that the table or schema does not exist. In this case, perform the following operations to grant permissions: 1. Log in to the Oracle node as user **root**. 2. Run the following command to switch to user **oracle**: **su oracle** 3. Run the following command to log in to the database: **sqlplus /nolog** 4. Run the following command to log in as user **sys**: **connect sys as sysdba;** Enter the password of user **sys**. 5. Run the following SQL statement to grant permissions: **GRANT SELECT ON GV_$INSTANCE to *xxx*;** In the preceding command, *xxx* indicates the name of the user to which the permissions will be granted.
Password	Yes	Password of the username

Parameter	Mandatory	Description
Connection Type	Yes	Select a connection type. • SID SID indicates the ID of the Oracle database instance. One instance corresponds to only one database, but one database can correspond to multiple instances. • Service Name It was introduced since Oracle8i and indicates the external service name of the Oracle database.
SID	Yes	This parameter is mandatory when **Connection type** is set to **SID**. SID indicates the ID of the Oracle database instance. One instance corresponds to only one database, but one database can correspond to multiple instances.
Service Name	Yes	This parameter is mandatory when **Connection type** is set to **Service Name**. This parameter was introduced since Oracle8i and indicates the external service name of the Oracle database.

3.2.16 Configuring a DIS Connection

Table 3-18 DIS connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**DIS** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.

Parameter	Mandatory	Description
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
Destination Project ID	Yes	The ID of the project that the destination DIS stream belongs to. The **DIS Client** node is used to send messages to the destination DIS stream.
Destination Region	Yes	Region that the target DIS stream belongs to. The DIS Client node is used to send messages to the target DIS stream.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source
Data Source Authentication and Other Function Configuration		
AK	Yes	The AK of the tenant who creates the destination DIS stream that receives messages from the **DIS Client** node.
SK	Yes	The SK of the tenant who creates the destination DIS stream that receives messages from the **DIS Client** node.
Description	No	Description of the connection

3.2.17 Configuring a Host Connection

Table 3-19 Host Connection parameters

Parameter	Mandatory	Description
Data Connection Type	Yes	**Host Connection** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).

Parameter	Mandatory	Description
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
Host Address	Yes	IP address of the Linux host For details, see **Viewing Details About an ECS**.
Agent	Yes	CDM cluster used as an agent. **NOTE** • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload. • When scheduling shell or Python scripts, the agent accesses the ECS. If shell and Python scripts are scheduled frequently, the ECS adds the private IP address of the agent to the blocklist. To ensure normal job scheduling, you are advised to use the **root** user of the ECS to add the private IP address bound to the agent (CDM cluster) to the **/etc/hosts.allow** file. For details about how to obtain the private IP address of the CDM cluster, see **Viewing Basic Cluster Information and Modifying Cluster Configurations**.
Port	Yes	SSH port number of the host. By default, port 22 is used to log in to a Linux host. If the port number has been changed, you can obtain the new port number from the **port** field in the **/etc/ssh/sshd_config** file.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source
Data Source Authentication and Other Function Configuration		

Parameter	Mandatory	Description
Username	Yes	Username for logging in to the host
Login Mode	Yes	Mode for logging in to the host • Key Pair • Password
Key Pair	Yes	This parameter is available only when **Login Mode** is set to **Key Pair**. If **Key Pair** is the login mode of the host, you need to obtain the private key file, upload it to OBS, and select an OBS path. **NOTE** The uploaded private key must match the public key configured on the host. For details, see **Application Scenarios for Using Key Pairs**.
Key Pair Password	Yes	If no password is set for the key pair, you do not need to set this parameter.
Password	Yes	This parameter is available only when **Login Mode** is set to **Password**. If the login mode of the host is to use a password, enter a login password.
Host Connection Description	No	Descriptive information about the host connection

NOTICE

- The maximum number of shell or Python scripts that can run concurrently on the ECS is determined by the value of **MaxSessions** in the **/etc/ssh/sshd_config** file on the ECS. Set **MaxSessions** based on the scheduling frequency of shell or Python scripts.
- You have the permission to create and execute files in the **/tmp** directory on the host.
- Shell and Python scripts are executed in the **/tmp** directory on an ECS. Ensure that the disk space of the **/tmp** directory is not used up.

3.3 Migrating Resources

To migrate resources in one workspace to another, you can use the resource migration function provided by DataArts Studio.

Resources that can be migrated include those in DataArts DataService and DataArts Catalog as well as the data connections in Management Center.

Prerequisites

- Resources can be imported from OBS or a local path.
- There are resources that can be migrated. For details on how to create data connections, see **Managing Data Connections**. For details on how to classify metadata and add tags, see **Tags**. For details on how to create collection tasks, see **Task Management**. For details on how to publish APIs, see **Publishing an API**.

Constraints

- Collection tasks with the same name cannot be migrated repeatedly.
- Categories and tags with the same name cannot be migrated repeatedly.
- Only an exported .zip file can be imported. During the import, the system verifies the resources in the file.
- For security concerns, passwords of connections are not exported when the connections are exported. You need to enter the passwords when importing the connections.
- Only the enterprise edition supports the export of data catalogs (categories, tags, and collection tasks). The expert edition does not support this function.
- The size of the file to be imported from an OBS bucket or local path cannot exceed 10 MB.

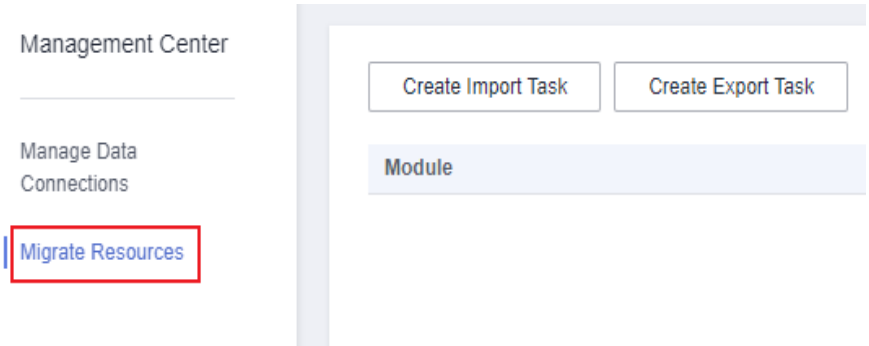
Exporting a Resource

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

Step 2 On the DataArts Studio console, locate a workspace and click **Management Center**.

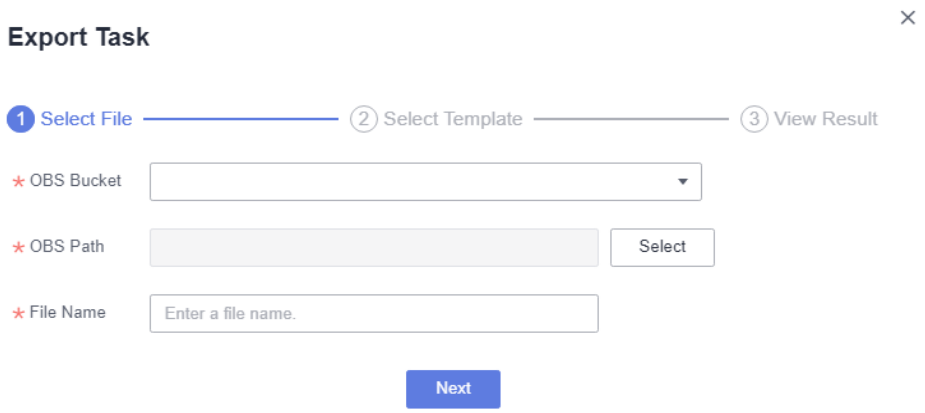
Step 3 In the navigation pane, choose **Migrate Resources**.

Figure 3-2 Migrating Resources



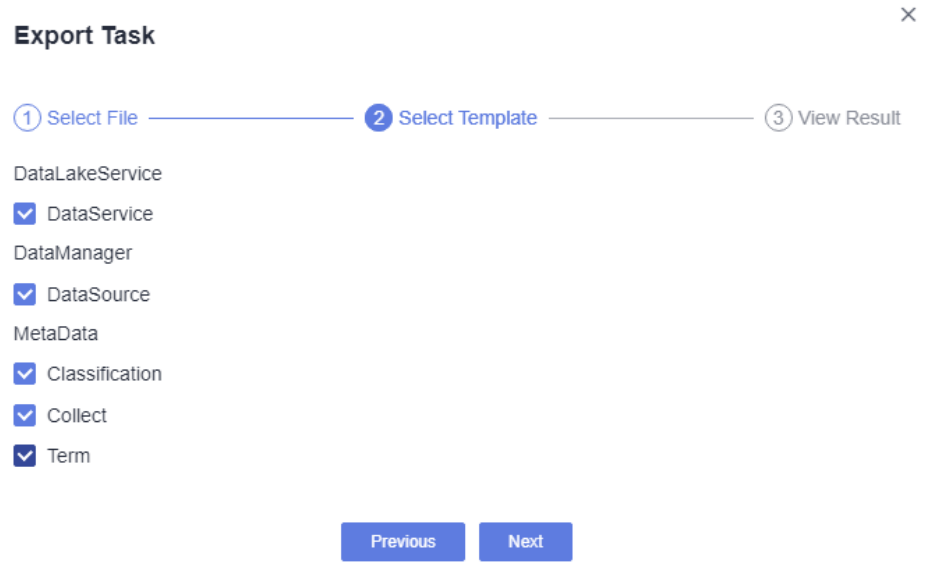
Step 4 Click **Create Export Task** to configure the file name and the OBS path for saving the file.

Figure 3-3 Export Task



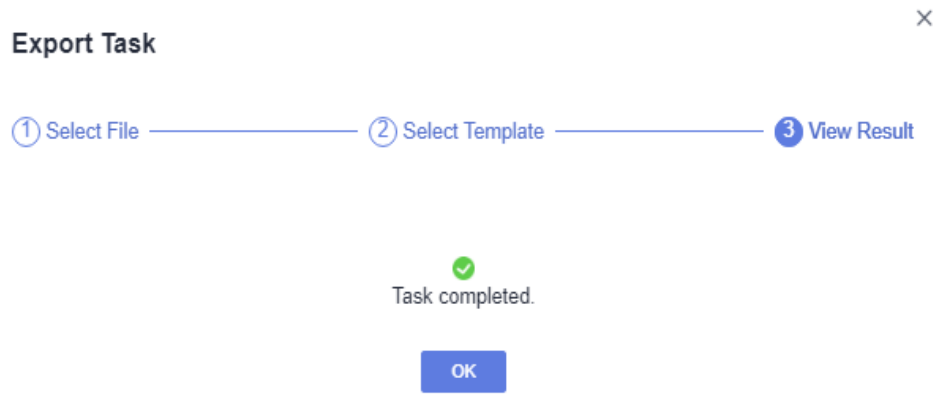
Step 5 Click **Next** and select the resources to export.

Figure 3-4 Selecting the resource to export



Step 6 Click **Next** and wait until the export is complete. The resource package is exported to the OBS path you have set.

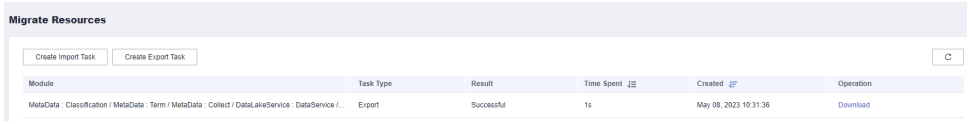
Figure 3-5 Export completed



If no result is displayed in 1 minute, the export fails. Try again. If the failure persists, contact the customer service or technical support.

Step 7 After the export is complete, you can click **Download** in the row of the corresponding migration task to download the exported resource package.

Figure 3-6 Downloading the exported result



----End

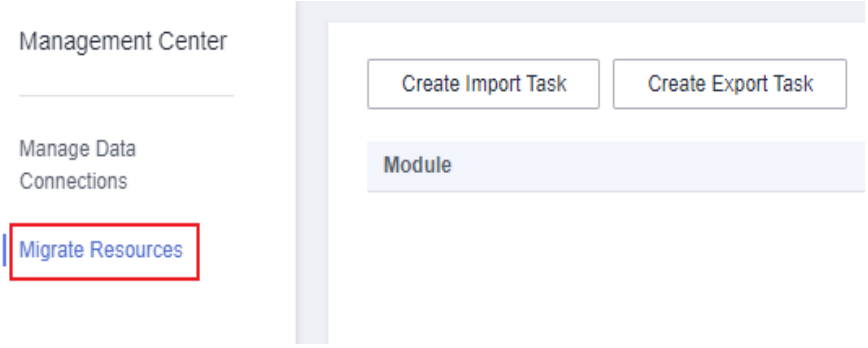
Importing a Resource

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

Step 2 On the DataArts Studio console, locate a workspace and click **Management Center**.

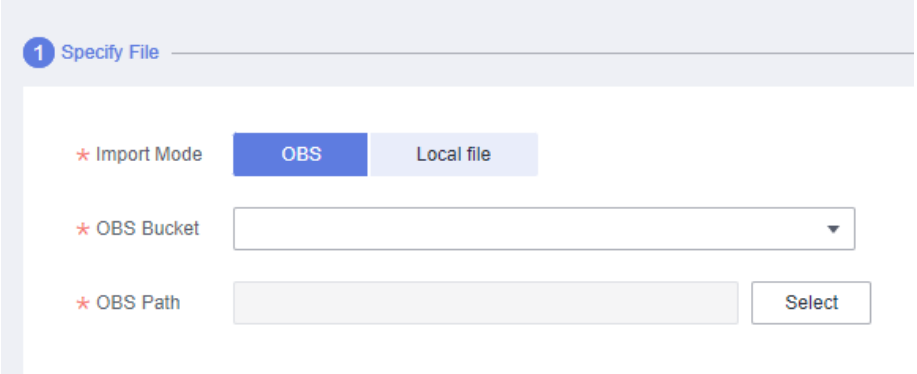
Step 3 In the navigation pane, choose **Migrate Resources**.

Figure 3-7 Migrating Resources



Step 4 Click **Create Import Task**. On the displayed page, select an import mode and set the OBS bucket and path or local path that stores resources. The resource to be imported must be a .zip file exported from the console.

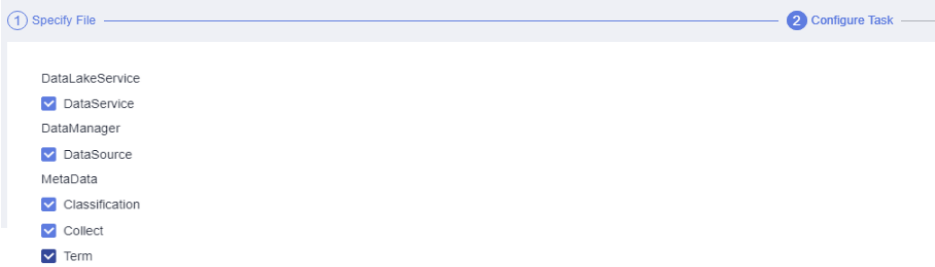
Figure 3-8 Configuring the path that stores the resources to be imported



Step 5 Click **Create Import Task** and upload a .zip resource file that you have exported.

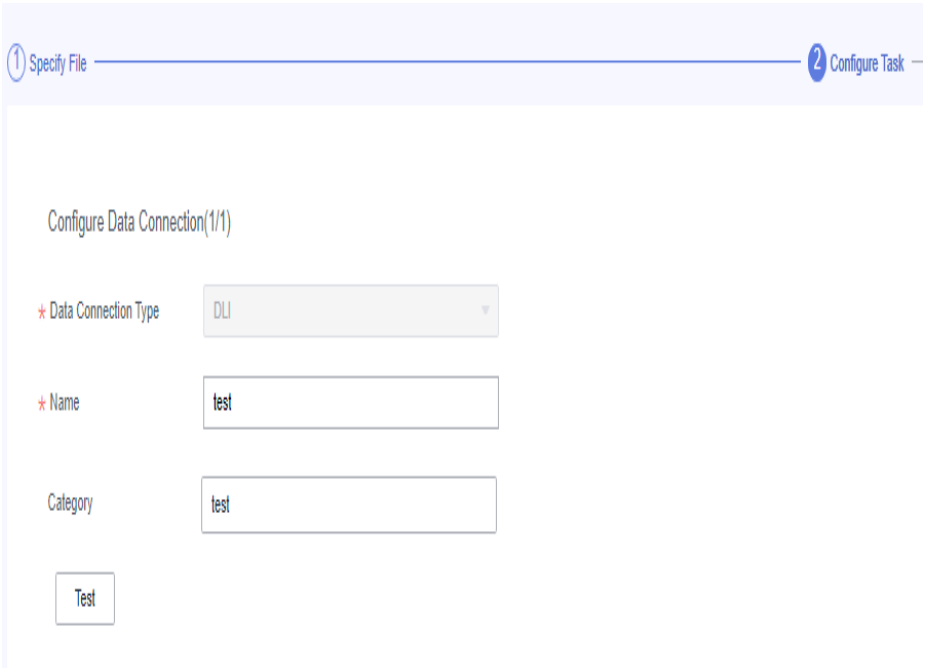
Step 6 Click **Next** and select the resources to import.

Figure 3-9 Selecting the resource to import



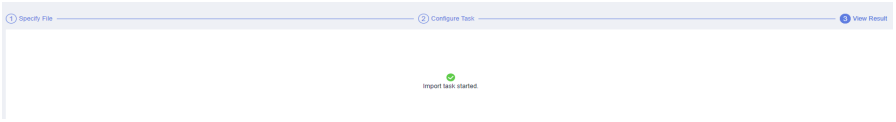
Step 7 If you select **DataSource**, click **Next** to configure a data connection.

Figure 3-10 Configuring a data connection



Step 8 Click **Next** and wait until the import task is delivered. When the import task is delivered successfully, the system displays message "Import task started."

Figure 3-11 Import task started



Step 9 Click **OK**. You can view the import result in the resource migration task list.

Subtasks that fail are marked in red. You can click their names to view the failure causes.

Figure 3-12 Viewing the import result

----End

3.4 Configuring Environment Isolation for a Workspace in Enterprise Mode

- You can configure isolation between the development and production environments for DLI and DB.

- After the environment isolation is configured, the data connection in the development environment in the script or job during data development is automatically switched to the data connection in the production environment after the process is released.

Prerequisites

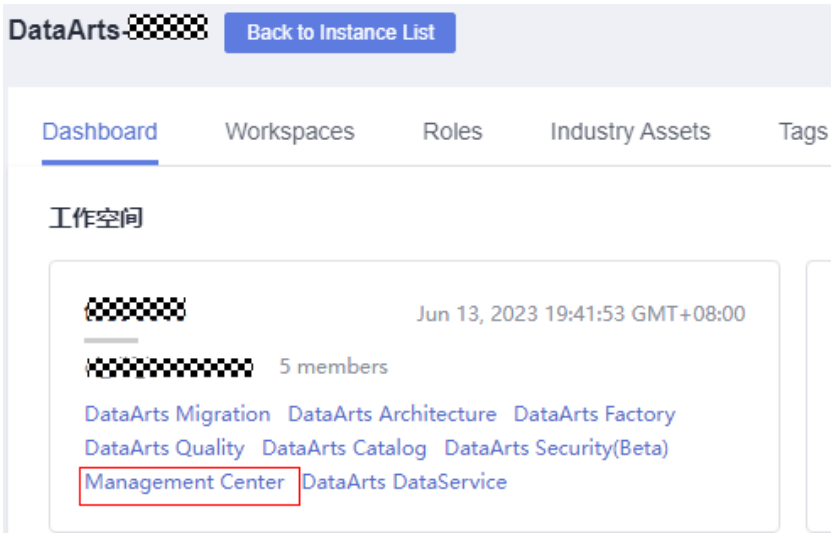
- Before configuring environment isolation for DLI, ensure that you have created a DLI **data connection**.

(Optional) Configuring DLI Environment Isolation

Environment isolation needs to be configured only for a serverless service (that is, DLI).

1. On the DataArts Studio console, locate a workspace and click **Management Center**.

   Figure 3-13 Management Center

2. In the left navigation pane on the **Management Center** page, choose **Data Source Resource Mapping Configuration**.

   Figure 3-14 Data Source Resource Mapping Configuration

3. Click the **DB Configuration** tab and then **Add**. Set the database names for the development and production environments respectively and click **Save**.

   You can click  and  to edit and delete records.

   The database names must be the names of created databases. It is recommended that the database name for the development environment be the same as that for the production environment, and that suffix **_dev** be

added to the database name for the development environment so that it can be distinguished from the database name for the production environment.

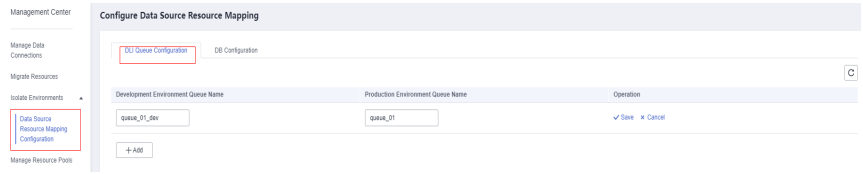
Figure 3-15 DB Configuration

4. Click the **DLI Queue Configuration** tab and then **Add**. Set the queue names for the development and production environments respectively and click **Save**.

   You can use  and  to edit and delete records.

   The queue names must be the names of created DLI queues. It is recommended that the queue name for the development environment be the same as that for the production environment, and that suffix **_dev** be added to the queue name for the development environment so that it can be distinguished from the queue name for the production environment.

Figure 3-16 DLI Queue Configuration



5. After the preceding operations are complete, DLI environment isolation configuration is complete.

DB Configuration

1. On the DataArts Studio console, locate a workspace and click **Management Center**.
2. In the left navigation pane on the **Management Center** page, choose **Data Source Resource Mapping Configuration**.
3. Click the **DB Configuration** tab and then **Add**. Set the database names for the development and production environments respectively and click **Save**.

   You can click  and  to edit and delete records.

   The database names must be the names of created databases. It is recommended that the database name for the development environment be the same as that for the production environment, and that suffix **_dev** be added to the database name for the development environment so that it can be distinguished from the database name for the production environment.

NOTICE

For GaussDB(DWS), MRS Hive, and MRS Spark, if you **select the same cluster** when creating a data connection, you must configure database mapping on the **Configure Data Source Resource Mapping** page to isolate the development and production environments.

Figure 3-17 DB Configuration

3.5 Tutorials

3.5.1 Creating an MRS Hive Connection

This section describes how to create an MRS Hive connection between DataArts Studio and the data lake base.

Prerequisites

- You have created a data lake to connect, for example, a database or cloud service supported by DataArts Studio.
  - Before creating a DWS data connection, ensure that you have created a cluster in DWS and have the permissions required to view Key Management Service (KMS) keys.
  - Before creating an MRS connection such as an MRS HBase and MRS Hive connection, ensure that you have bought an MRS cluster and selected required components.
- The data lake to connect communicates with the DataArts Studio instance properly.
  - If the data lake is an on-premises database, a public network or a dedicated connection is required. Ensure that the host where the data source is located can access the public network and the port has been enabled in the firewall rule.
  - If the data lake is a cloud service (such as DWS and MRS), the following requirements must be met for network interconnection:
    - If the CDM cluster in the DataArts Studio instance and the cloud service are in different regions, a public network or a dedicated connection is required.
    - If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same

VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**.

- The cloud service instance and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.

- If the enterprise mode is used, pay attention to the following points:

  In enterprise mode, the development environment and production environment need to be distinguished. Therefore, you need to prepare two sets of data lake services for the production environment and development environment to isolate the development environment from the production environment.

  - **If two clusters are used** for clustered data sources, such as MRS, GaussDB(DWS), RDS, MySQL, Oracle, DIS, and ECS, you can create data connections in Management Center to distinguish data lake services in the development environment from those in the production environment. The data lake is automatically switched during development and production. Therefore, you need to prepare two sets of data lake services. The versions, specifications, components, regions, VPCs, subnets, and related configurations of the two sets of data lake services must be the same. For details on how to create data connections, see **Creating a Data Connection**.
  - For serverless services (such as DLI), DataArts Studio configures the mapping between data lake services in the production environment and development environment through environment isolation in the management center. The corresponding data lake is automatically switched during the development and production processes. Therefore, you need to prepare two sets of queues and database resources in the serverless data lake service and distinguish them by name suffix. For details, see **Configuring Environment Isolation for a Workspace in Enterprise Mode**.
  - For GaussDB(DWS), MRS Hive, and MRS Spark, if you **select the same cluster** when creating a data connection, you must configure database mapping on the **Configure Data Source Resource Mapping** page to isolate the development and production environments. For details, see **DB configuration**.

  For example, if your data lake service is an MRS cluster, you need to prepare two MRS clusters with the same version, specifications, components, region, VPC, and subnet. If some configurations of an MRS cluster are modified, you also need to synchronize the modifications to the other MRS cluster.

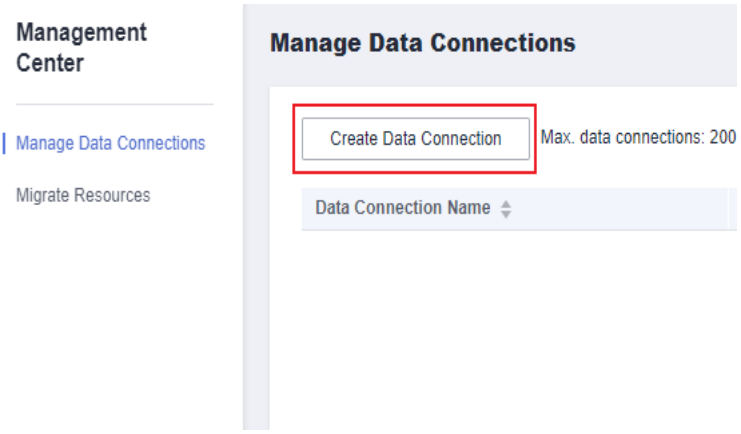
Creating a Data Connection

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

Step 2 On the DataArts Studio console, locate a workspace and click **Management Center**.

Step 3 On the displayed **Manage Data Connections** page, click **Create Data Connection**.

Figure 3-18 Creating a data connection



Step 4 On the **Manage Data Connections** page, click **Create Data Connection**. On the displayed page, select **MRS Hive** for **Data Connection Type** and set other parameters based on the descriptions in **Table 3-20**.

Figure 3-19 MRS Hive connection parameters

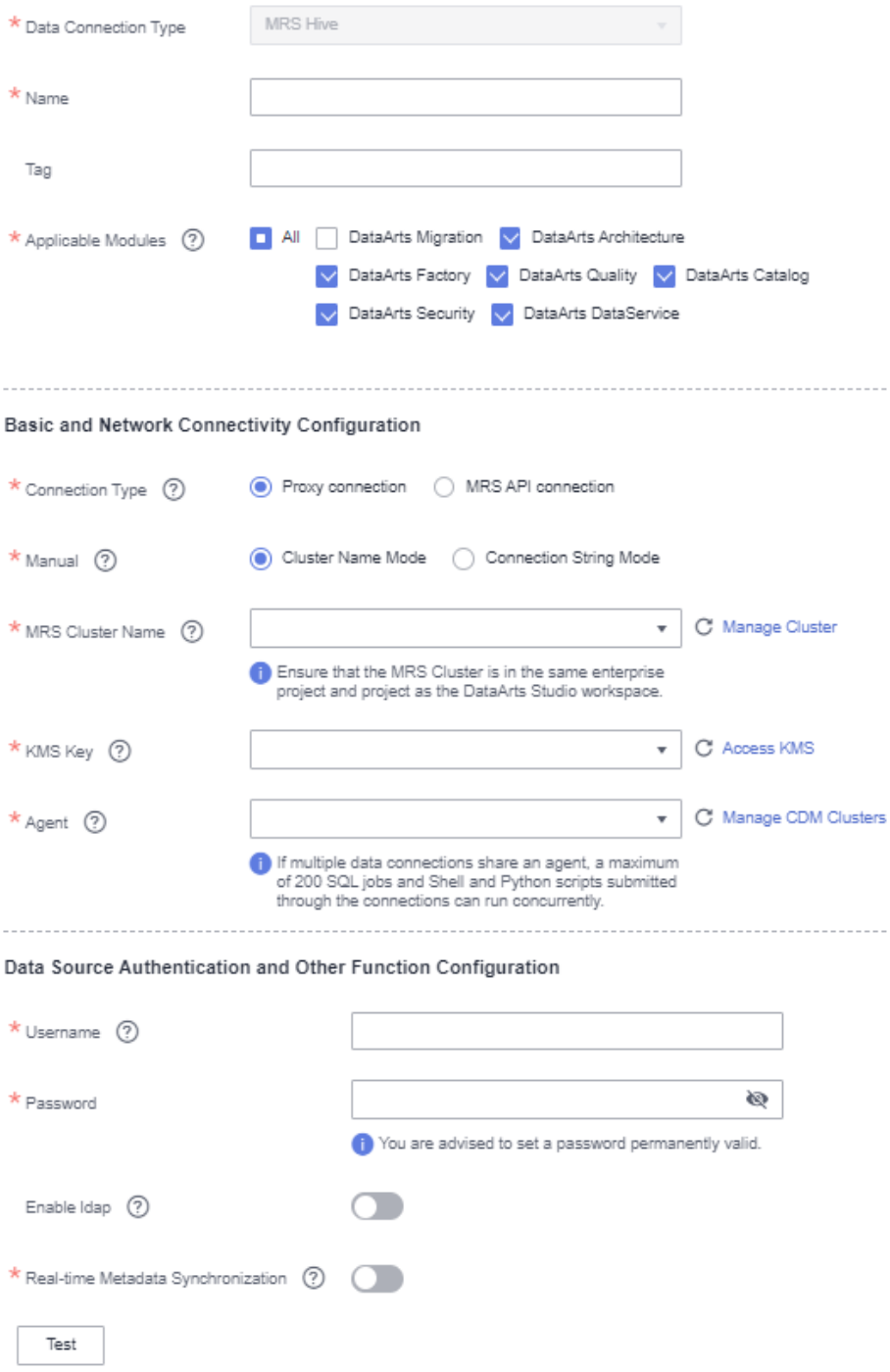


Table 3-20 MRS Hive connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**MRS Hive** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		

Parameter	Mandatory	Description
Connection Type	Yes	Connection type. **Proxy connection** is recommended. • **Proxy connection**: An agent (CDM cluster) is used to access MRS clusters. This method supports all versions of MRS clusters. • **MRS API connection**: MRS APIs are used to access MRS clusters. This method supports only MRS clusters of the 2.*X* or a later version. When you select **MRS API connection**, pay attention to the following restrictions: 1. The MRS API connection is available only for DataArts Factory. 2. In DataArts Factory, you cannot view or manage the databases, data tables, and fields of the connection in a visualized manner. If an MRS cluster of version 3.2.1 or later is connected, you can view rather than manage the databases, data tables, and fields of the connection in a visualized manner. 3. When the SQL editor of DataArts Factory is used to run SQL statements, the execution results can be displayed only in logs. **NOTE** Select **Proxy connection** for **Connection Type** so that the DataArts Architecture, DataArts Quality, DataArts Catalog, and DataArts DataService components can use the MRS connection.
Manual	Yes	This parameter is mandatory when **Connection Type** is set to **Proxy connection**. Select the connection mode. If you do not need to access MRS clusters in other projects or enterprise projects, select **Cluster Name Mode**. • **Cluster Name Mode**: Select an existing cluster. You can only connect to an MRS cluster in the same project and enterprise project. • If you select **Connection String Mode**, you can set **Manager IP** and enable communication between this connection's agent (CDM cluster) and an MRS cluster in another project or enterprise project so that you can access the MRS cluster.

Parameter	Mandatory	Description
Manager IP	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. Set this parameter to the floating IP address of MRS Manager. Only MRS clusters are supported. A Hadoop cluster can be connected only after it is managed by MRS. You can click **Select** next to the text box and select an MRS cluster in the same project and enterprise project. If you want to access an MRS cluster in another project or enterprise project, obtain and enter the floating IP address of MRS Manager and ensure that the connection's agent (CDM cluster) can communicate with the tenant-plane MRS cluster. To obtain the floating IP address of MRS Manager, log in to the active master node of the MRS cluster and run the **ifconfig** command. In the command output, the IP address of **eth0:wsom** is the floating IP address of MRS Manager. For details about how to log in to the master node of the MRS cluster, see **Logging In to an ECS**. Enter multiple IP addresses based on the scenario in sequence and separate them with commas (,), for example, **127.0.0.1** or **127.0.0.1,127.0.0.2,127.0.0.3**. • If you enter one IP address, enter the management-plane floating IP address of the MRS cluster. • If you enter three IP addresses, enter the IP address of the active node on the MRS cluster service plane, IP address of the standby node on the MRS cluster service plane, and the floating IP address of the MRS cluster management plane.

Parameter	Mandatory	Description
MRS Cluster Name	Yes	This parameter is mandatory when **MRS API connection** is selected for **Connection Type** or **Cluster Name Mode** is selected for **Manual**. The name of the MRS cluster. Select an MRS cluster that Hive belongs to. Only MRS clusters are supported. A Hadoop cluster can be selected only after it is managed by MRS. All the MRS clusters with the same project ID and enterprise project are displayed. If the connection fails after you select a cluster, check whether the MRS cluster can communicate with the CDM instance which functions as the agent. They can communicate with each other in the following scenarios: • If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule. • If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**. • The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.
KMS Key	No	This parameter is mandatory when **Connection Type** is set to **Proxy connection**. KMS key used to encrypt and decrypt the authentication information for the data source

Parameter	Mandatory	Description
Agent	Yes	This parameter is mandatory when **Connection Type** is set to **Proxy connection**. MRS is not a fully managed service and cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating an MRS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with the MRS cluster. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as the MRS cluster. The security group rule must also allow the CDM cluster communicate with the MRS cluster. **NOTE** • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster cannot connect to multiple MRS security clusters. You are advised to plan multiple agents which are mapped to MRS security clusters one by one. • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
Data Source Authentication and Other Function Configuration		
Authentication Method	Yes	This parameter is mandatory when **Connection String Mode** is selected for **Manual**. It specifies the authentication method used for accessing the MRS cluster. The following options are available: • **SIMPLE**: for non-security mode • **KERBEROS**: for security mode

Parameter	Mandatory	Description
Username	Yes	Human-machine user of the MRS cluster. This parameter is mandatory when **Connection Type** is set to **Proxy connection**. If a new MRS user is used for connection, you need to log in to Manager and change the initial password. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user whose password never expires by referring to **Creating a Kerberos Authentication User for an MRS Security Cluster**. When creating an MRS data connection, set **Username** and **Password** to the new MRS username and password. **NOTE** • For clusters of MRS 3.1.0 or later, the user must at least have permissions of the **Manager_viewer** role to create data connections in Management Center. To perform database, table, and data operations on components, the user must also have user group permissions of the components. • For clusters earlier than MRS 3.1.0, the user must have permissions of the **Manager_administrator** or **System_administrator** role to create data connections in Management Center. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections. • You are advised to set a user password that never expires to prevent connection failures and service loss caused by password expiration.
Password	Yes	The password for accessing the MRS cluster. This parameter is mandatory when **Connection Type** is set to **Proxy connection**.
Enable ldap	No	This parameter is available when **Connection Type** is set to **Proxy connection**. If LDAP authentication is enabled for an external LDAP server connected to MRS Hive, the LDAP username and password are required for authenticating the connection to MRS Hive. In this case, this option must be enabled. Otherwise, the connection will fail.
ldapUsername	Yes	This parameter is mandatory when **Enable ldap** is enabled. Enter the username configured when LDAP authentication was enabled for MRS Hive.

Parameter	Mandatory	Description
ldapPassword	Yes	This parameter is mandatory when **Enable ldap** is enabled. Enter the password configured when LDAP authentication was enabled for MRS Hive.

Step 5 Click **Test** to test connectivity of the data connection. If the test fails, the data connection fails to be created.

Step 6 After the test is successful, click **OK** to create the data connection.

----End

Reference

1. Why is no MRS Hive cluster displayed on the Create Data Connection page?

   Possible causes are as follows:

   - Hive/HBase components were not selected during MRS cluster creation.
   - The network between the CDM cluster and MRS cluster was disconnected when an MRS data connection is created.

     The CDM cluster functions as a network agent. MRS data connections that you are going to create need to communicate with CDM.

2. Why does a Hive data connection fail to obtain information about databases or tables?

   The possible cause is that the CDM cluster is stopped or a concurrency conflict occurs. You can switch to another agent to temporarily avoid this issue.

3.5.2 Creating a DWS Connection

This section describes how to create a DWS connection between DataArts Studio and the data lake base.

Prerequisites

- You have created a data lake to connect, for example, a database or cloud service supported by DataArts Studio.
  - Before creating a DWS data connection, ensure that you have created a cluster in DWS and have the permissions required to view Key Management Service (KMS) keys.
  - Before creating an MRS connection such as an MRS HBase and MRS Hive connection, ensure that you have bought an MRS cluster and selected required components.
- The data lake to connect communicates with the DataArts Studio instance properly.
  - If the data lake is an on-premises database, a public network or a dedicated connection is required. Ensure that the host where the data

source is located can access the public network and the port has been enabled in the firewall rule.

- If the data lake is a cloud service (such as DWS and MRS), the following requirements must be met for network interconnection:
  - If the CDM cluster in the DataArts Studio instance and the cloud service are in different regions, a public network or a dedicated connection is required.
  - If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**.
  - The cloud service instance and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.

- If the enterprise mode is used, pay attention to the following points:

  In enterprise mode, the development environment and production environment need to be distinguished. Therefore, you need to prepare two sets of data lake services for the production environment and development environment to isolate the development environment from the production environment.

  - **If two clusters are used** for clustered data sources, such as MRS, GaussDB(DWS), RDS, MySQL, Oracle, DIS, and ECS, you can create data connections in Management Center to distinguish data lake services in the development environment from those in the production environment. The data lake is automatically switched during development and production. Therefore, you need to prepare two sets of data lake services. The versions, specifications, components, regions, VPCs, subnets, and related configurations of the two sets of data lake services must be the same. For details on how to create data connections, see **Creating a Data Connection**.
  - For serverless services (such as DLI), DataArts Studio configures the mapping between data lake services in the production environment and development environment through environment isolation in the management center. The corresponding data lake is automatically switched during the development and production processes. Therefore, you need to prepare two sets of queues and database resources in the serverless data lake service and distinguish them by name suffix. For details, see **Configuring Environment Isolation for a Workspace in Enterprise Mode**.
  - For GaussDB(DWS), MRS Hive, and MRS Spark, if you **select the same cluster** when creating a data connection, you must configure database mapping on the **Configure Data Source Resource Mapping** page to isolate the development and production environments. For details, see **DB configuration**.

For example, if your data lake service is an MRS cluster, you need to prepare two MRS clusters with the same version, specifications, components, region, VPC, and subnet. If some configurations of an MRS cluster are modified, you also need to synchronize the modifications to the other MRS cluster.

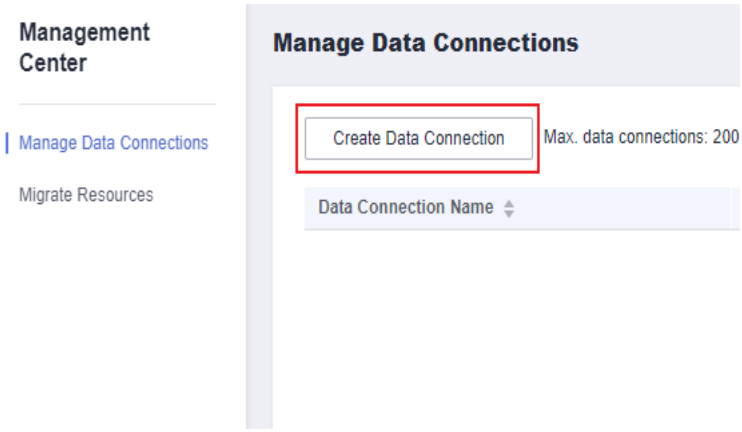
Creating a Data Connection

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

Step 2 On the DataArts Studio console, locate a workspace and click **Management Center**.

Step 3 On the displayed **Manage Data Connections** page, click **Create Data Connection**.

Figure 3-20 Creating a data connection



Step 4 On the **Manage Data Connections** page, click **Create Data Connection**. On the displayed page, select **DWS** for **Data Connection Type** and set other parameters based on the descriptions in **Table 3-21**.

Figure 3-21 DWS connection parameters

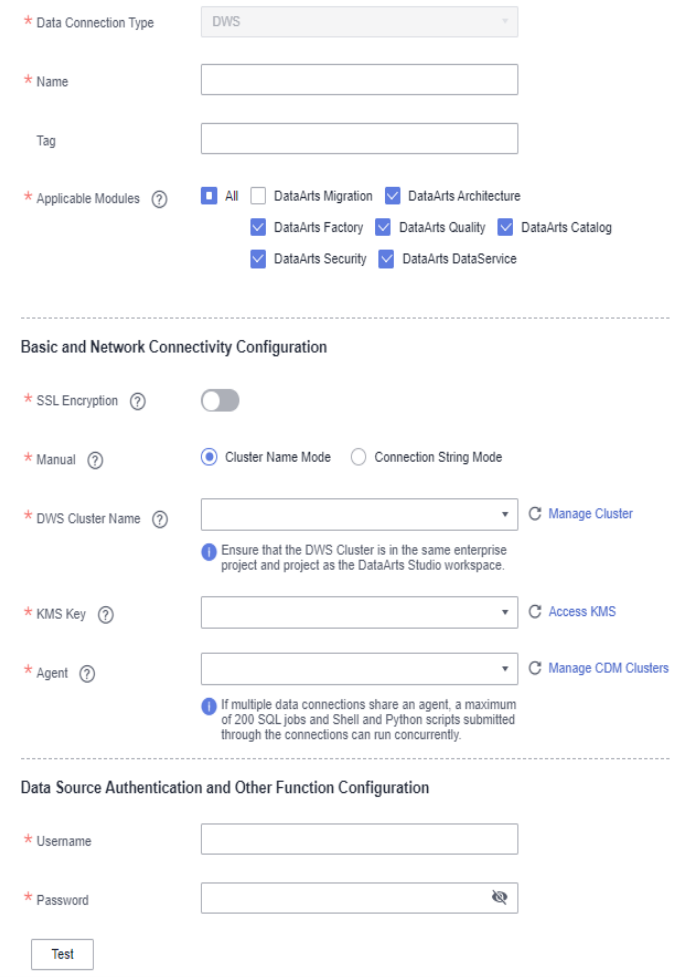


Table 3-21 DWS connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**DWS** is selected and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).

Parameter	Mandatory	Description
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
SSL Encryption	Yes	DWS supports SSL encryption and certificate authentication for communication between the client and server. You can use **SSL Connection** to set the communication mode. If **SSL Connection** is enabled, only SSL encryption can be used. If **SSL Connection** is disabled, both modes can be used. This function is disabled by default.
Manual	Yes	Select either of the following modes: • **Cluster Name Mode**: Select an existing cluster. • **Connection String Mode**: Enter the IP address/ domain name and port of the corresponding cluster and enable the communication between the connection's agent (CDM cluster) and the DWS cluster.
DWS Cluster Name	Yes	This parameter is mandatory when **Manual** is set to **Cluster Name Mode**. Select a DWS cluster from all the DWS clusters with the same project ID and enterprise project.

Parameter	Mandatory	Description
IP Address or Domain Name	Yes	This parameter is mandatory when **Manual** is set to **Connection String Mode**. This parameter indicates the address for accessing the cluster database through an internal network. Enter an IP address or domain name. The IP address or domain name is automatically generated during cluster creation. You can obtain them on the management console by performing the following operations: 1. Log in to the GaussDB(DWS) console. 2. In the left navigation pane, choose **Instances**. 3. Click the name of an instance to enter the basic information page. In the **Connection Information** area, you can obtain the private IP address, domain name, and port number.
Port	Yes	This parameter is mandatory when **Manual** is set to **Connection String Mode**. This parameter indicates the database port number specified during the DWS cluster creation. Ensure that you have enabled this port in the security group rule so that the DataArts Studio instance can connect to the database in the DWS cluster through this port.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source

Parameter	Mandatory	Description
Agent	Yes	Data Warehouse Service (DWS) is not a fully managed service and thus cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating a DWS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with the DWS cluster. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as the DWS cluster. The security group rule must also allow the CDM cluster communicate with the DWS cluster. **NOTE** If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
Data Source Authentication and Other Function Configuration		
Username	Yes	The database username, which is specified when the DWS cluster is created.
Password	Yes	The password for accessing the database, which is specified when the DWS cluster is created.

Step 5 Click **Test** to test connectivity of the data connection. If the test fails, the data connection fails to be created.

Step 6 After the test is successful, click **OK** to create the data connection.

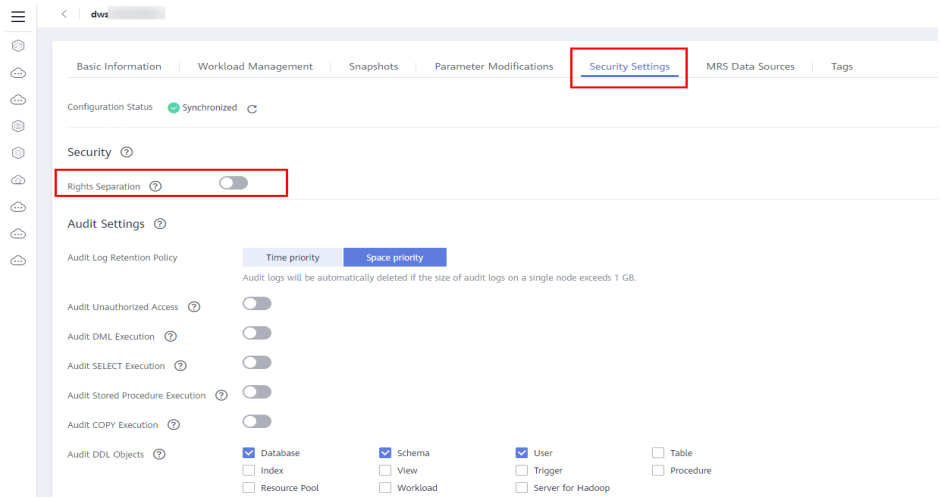
----End

Reference

1. What should I do if the connection test fails when I enable the SSL connection during the creation of a DWS data connection?

   The failure may be caused by the rights separation function of the DWS cluster. On the DWS console, click the corresponding cluster, choose **Security Settings**, and disable **Rights Separation**.

Figure 3-22 Disabling Rights Separation for the DWS cluster



2. Why does a DWS data connection fail to obtain information about databases or tables?

   The possible cause is that the CDM cluster is stopped or a concurrency conflict occurs. You can switch to another agent to temporarily avoid this issue.

3.5.3 Creating a MySQL Connection

This section describes how to create a MySQL connection between DataArts Studio and the data lake base.

Prerequisites

- You have created a data lake to connect, for example, a database or cloud service supported by DataArts Studio.
  - Before creating a DWS data connection, ensure that you have created a cluster in DWS and have the permissions required to view Key Management Service (KMS) keys.
  - Before creating an MRS connection such as an MRS HBase and MRS Hive connection, ensure that you have bought an MRS cluster and selected required components.
- The data lake to connect communicates with the DataArts Studio instance properly.
  - If the data lake is an on-premises database, a public network or a dedicated connection is required. Ensure that the host where the data source is located can access the public network and the port has been enabled in the firewall rule.
  - If the data lake is a cloud service (such as DWS and MRS), the following requirements must be met for network interconnection:
    - If the CDM cluster in the DataArts Studio instance and the cloud service are in different regions, a public network or a dedicated connection is required.
    - If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same

VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**.

- The cloud service instance and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.

- If the enterprise mode is used, pay attention to the following points:

  In enterprise mode, the development environment and production environment need to be distinguished. Therefore, you need to prepare two sets of data lake services for the production environment and development environment to isolate the development environment from the production environment.

  - **If two clusters are used** for clustered data sources, such as MRS, GaussDB(DWS), RDS, MySQL, Oracle, DIS, and ECS, you can create data connections in Management Center to distinguish data lake services in the development environment from those in the production environment. The data lake is automatically switched during development and production. Therefore, you need to prepare two sets of data lake services. The versions, specifications, components, regions, VPCs, subnets, and related configurations of the two sets of data lake services must be the same. For details on how to create data connections, see **Creating a Data Connection**.
  - For serverless services (such as DLI), DataArts Studio configures the mapping between data lake services in the production environment and development environment through environment isolation in the management center. The corresponding data lake is automatically switched during the development and production processes. Therefore, you need to prepare two sets of queues and database resources in the serverless data lake service and distinguish them by name suffix. For details, see **Configuring Environment Isolation for a Workspace in Enterprise Mode**.
  - For GaussDB(DWS), MRS Hive, and MRS Spark, if you **select the same cluster** when creating a data connection, you must configure database mapping on the **Configure Data Source Resource Mapping** page to isolate the development and production environments. For details, see **DB configuration**.

  For example, if your data lake service is an MRS cluster, you need to prepare two MRS clusters with the same version, specifications, components, region, VPC, and subnet. If some configurations of an MRS cluster are modified, you also need to synchronize the modifications to the other MRS cluster.

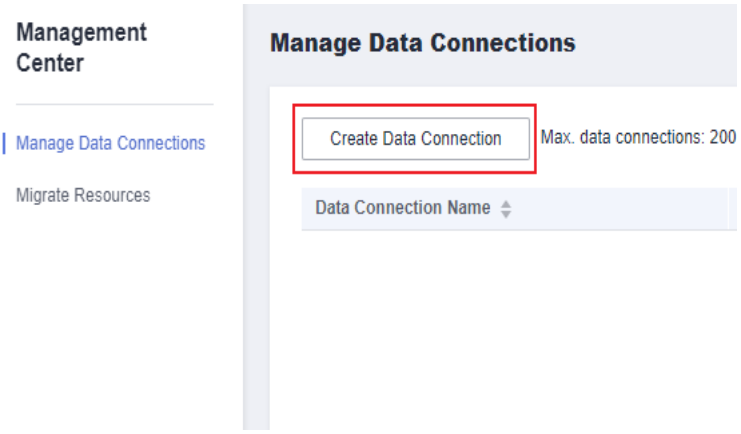
Creating a Data Connection

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

Step 2 On the DataArts Studio console, locate a workspace and click **Management Center**.

Step 3 On the displayed **Manage Data Connections** page, click **Create Data Connection**.

Figure 3-23 Creating a data connection



Step 4 On the **Manage Data Connections** page, click **Create Data Connection**. On the displayed page, select **RDS** for **Data Connection Type** and set other parameters based on the descriptions in **Table 3-22**.

NOTE

- You are not advised to select **MySQL (pending offline)** for **Data Connection Type**. Instead, You are advised to select **RDS**.
- RDS data connections depend on OBS. If OBS is unavailable in the same region as DataArts Studio, RDS data connections are not supported.

Figure 3-24 RDS connection parameters

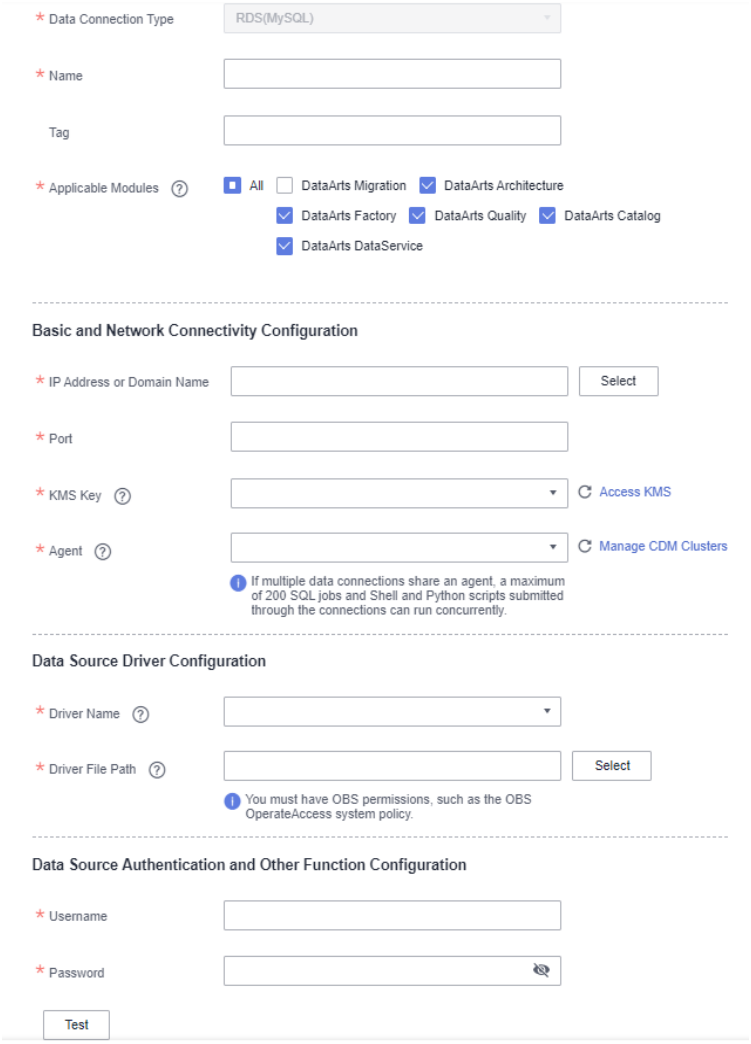


Table 3-22 RDS connection

Parameter	Mandatory	Description
Data Connection Type	Yes	**RDS** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).

Parameter	Mandatory	Description
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
IP Address or Domain Name	Yes	Address for accessing the relational database data source. The value can be an IP address or a domain name. • If the data source is RDS or GaussDB, you can obtain the address on the management console by performing the following operations: 1. Log in to the management console of the corresponding cloud service using the account you have obtained. 2. In the left navigation pane, choose **Instances**. 3. Click the name of an instance to enter the basic information page. In the **Connection Information** area, you can obtain the private IP address, domain name, and port number. **NOTE** Only the GaussDB data source supports multiple domain names. Use commas (,) to separate multiple domain names. • If the data source is MySQL, PostgreSQL, or DM, you can obtain the access address from the database administrator.

Parameter	Mandatory	Description
Port	Yes	Port for accessing the relational database. • If the data source is RDS or GaussDB, you can obtain the address on the management console by performing the following operations: 1. Log in to the management console of the corresponding cloud service using the account you have obtained. 2. In the left navigation pane, choose **Instances**. 3. Click the name of an instance to enter the basic information page. In the **Connection Information** area, you can obtain the private IP address, domain name, and port number. **NOTE** Only the GaussDB data source supports multiple domain names. Use commas (,) to separate multiple domain names. • If the data source is MySQL, PostgreSQL, or DM, you can obtain the access address from the database administrator.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source
Agent	Yes	RDS is not a fully managed service and cannot be directly connected to DataArts Studio. A CDM cluster can provide an agent for DataArts Studio to communicate with non-fully-managed services. Therefore, you need to select a CDM cluster when creating an RDS data connection. If no CDM cluster is available, create one first. As a network proxy, the CDM cluster must be able to communicate with RDS. To ensure network connectivity, the CDM cluster must be in the same region, AZ, VPC, and subnet as RDS. The security group rule must also allow the CDM cluster to communicate with RDS. **NOTE** If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload.
Data Source Driver Configuration		

Parameter	Mandatory	Description
Driver Name	Yes	The name of the driver. The following values are available: • **com.mysql.jdbc.Driver**: Select this driver name for RDS for MySQL or MySQL. • **org.postgresql.Driver**: Select this driver name for RDS for PostgreSQL or PostgreSQL. • **com.microsoft.sqlserver.jdbc.SQLServerDriver**: Select this driver name for RDS for SQL Server. • **dm.jdbc.driver.DmDriver**: Select this driver name for the Dameng database. • **com.huawei.opengauss.jdbc.Driver**: Select this driver name for RDS for GaussDB.
Driver File Path	Yes	It specifies the OBS path where the driver file is located. You need to download a .jar driver file from the corresponding official website and upload it to OBS. • MySQL driver: Download it from **https://downloads.mysql.com/archives/c-j/**. The 5.1.48 version is recommended. • PostgreSQL driver: Download it from **https://mvnrepository.com/artifact/org.postgresql/postgresql**. The 42.3.4 version is recommended. • SQL Server driver: Download it from **https://learn.microsoft.com/en-us/sql/connect/jdbc/download-microsoft-jdbc-driver-for-sql-server?view=sql-server-ver16**. The 8.4.1 version is recommended. • Dameng database driver: Obtain **DmJdbcDriver18.jar** from the DM installation directory **/dmdbms/drivers/jdbc**. • GaussDB driver: Search for "JDBC Package, Driver Class, and Environment Class" in *GaussDB User Guide* in the **GaussDB Documentation**, select the document corresponding to the instance version, and obtain the driver package by referring to the document. **NOTE** • The OBS path of the driver file cannot contain Chinese characters. • To update the driver, you must restart the CDM cluster in DataArts Migration and then edit the data connection to upload the driver.
Data Source Authentication and Other Function Configuration		

Parameter	Mandatory	Description
Username	Yes	The username of the database. The username is required for creating a cluster.
Password	Yes	The password for accessing the database. The password is required for creating a cluster.

Step 5 Click **Test** to test connectivity of the data connection. If the test fails, the data connection fails to be created.

Step 6 After the test is successful, click **OK** to create the data connection.

----End

Reference

1. What Are the Precautions for Creating an RDS Data Connection?

   When creating an RDS data connection, you need to bind an agent provided by the CDM cluster. Currently, a version of the CDM cluster earlier than 1.8.6 is not supported.

4 DataArts Migration

4.1 Overview

DataArts Migration is an efficient and easy-to-use data integration service. Based on the big data migration to the cloud and intelligent data lake solutions, CDM provides easy-to-use migration capabilities and can integrate various types of data sources into the data lake, which simplifies data source migration and integration and improves efficiency for you.

In this document, DataArts Migration refers to Cloud Data Migration (CDM).

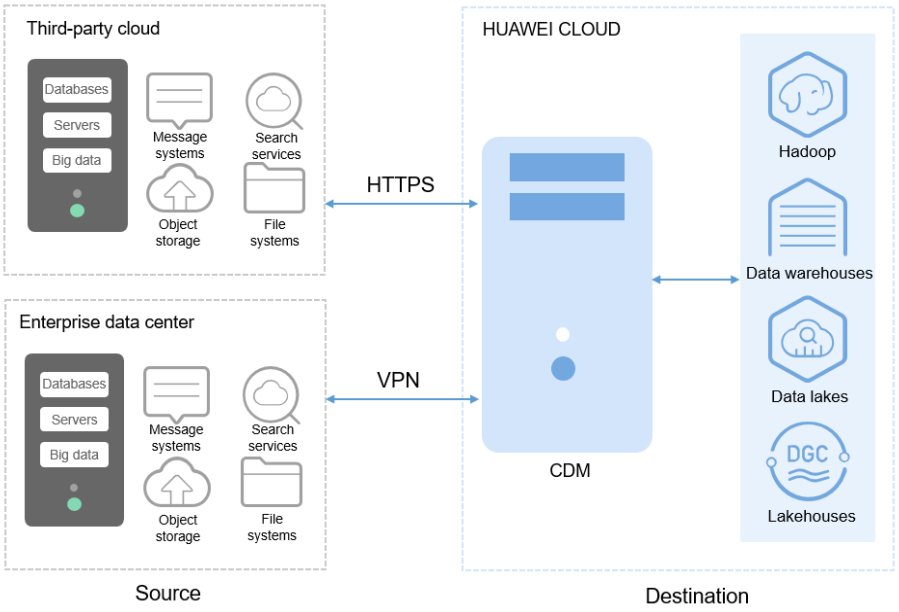
You can access the CDM console using either of the following methods:

- Log in to the CDM console and choose **Cluster Management** in the navigation pane.
- Log in to the DataArts Studio console. Locate a workspace and click **DataArts Migration**.

Introduction to CDM

CDM uses a distributed compute framework and concurrent processing techniques to help you migrate enterprise data in batches without any downtime and rapidly build desired data structures.

Figure 4-1 CDM



Functions

- **Table/file/entire DB migration**

  Tables or files can be migrated in batches. An entire database can be migrated between homogeneous and heterogeneous databases. A job can migrate hundreds of tables.

- **Incremental data migration**

  CDM supports incremental migration of files, relational databases, and HBase/CloudTable, as well as with WHERE clauses and macro variables of date and time.

- **Migration in transaction mode**

  When a CDM job fails to be executed, CDM rolls back the data to the state before the job starts and automatically deletes data from the destination table.

- **Field conversion**

  CDM supports field conversion functions, such as anonymization, character string operations, and date operations.

- **File encryption**

  When files are migrated to a file system, CDM can encrypt the files written to the cloud.

- **MD5 verification**

  MD5 verification is supported to check the file consistency from end to end and output verification result.

- **Dirty data archiving**

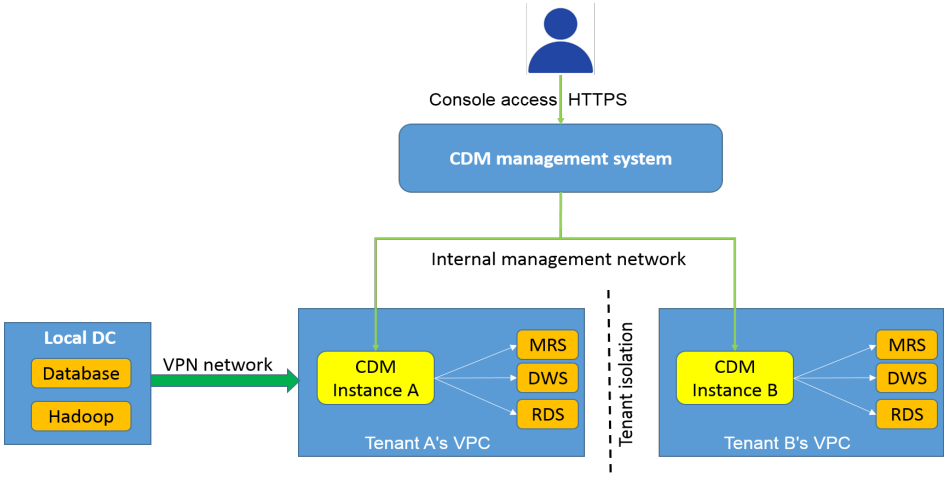
  CDM can archive the data that fails to be processed during migration, has been filtered out, or is not compliant with conversion or cleaning rules to dirty data logs. The threshold for dirty data ratio can be set to determine whether a task is successful.

Migration Principles

When a tenant uses CDM, the CDM system provisions a fully-managed CDM instance in the tenant's VPC. The instance allows only console and RESTful API access. Therefore the tenant cannot access the instance through other interfaces (such as SSH). This ensures data isolation between CDM tenants, prevents data leakage, and ensures transmission security during data migration between different cloud services in a VPC. Tenants can also use the VPN to migrate data from the on-premises data center to cloud services to ensure migration security.

CDM works in push-pull mode. CDM pulls data from the migration source and pushes the data to the migration destination. Data access operations are initiated by CDM. SSL will be used if the data source (such as RDS) supports it. During the migration, the usernames and passwords of the migration source and destination are required. Such information is stored in the database of the CDM instance. Protecting such information is critical to ensure CDM security.

Figure 4-2 Migration principles



4.2 Notes and Constraints

CDM System Notes and Constraints

1. A free CDM cluster provided together with the DataArts Studio instance can be used as an agent for the data connections in Management Center. However, you are not advised to use the cluster as a node for running data migration jobs when the cluster is used as an agent.
2. You can purchase CDM clusters of other specifications on the DataArts Studio console as incremental packages or directly purchase clusters on the CDM console. The differences are as follows:
   a. Billing: Clusters purchased on the DataArts Studio console only support packages purchased on the DataArts Studio console, while clusters purchased on the CDM console only support the discount packages purchased on the CDM console.
   b. Permission control: For clusters purchased on the DataArts Studio console, permissions are managed based on the DataArts Studio

permission system. For clusters purchased on the CDM console, permissions are managed based on the CDM permission system.

c. Application scenarios: Clusters purchased on the DataArts Studio console are isolated by workspace and can be used only in associated workspaces. Clusters purchased on the CDM console are not isolated by workspace and can be used in all DataArts Studio workspaces.

3. You cannot modify the flavor of an existing cluster. If you require a higher flavor, create a cluster with your desired flavor.
4. The CDM cluster version (Arm or x86) is determined by the architecture of underlying resources.
5. CDM does not support the function of controlling the data migration speed. Therefore, do not perform data migration during peak hours.
6. During data migration, CDM imposes pressure on the data source. You are advised to create a database account for data migration and configure an account policy to reduce the resource consumption of the data source. For example, you can configure a policy to delete the connections of the account when the CPU usage exceeds 30% to prevent impact on services.
7. The baseline and maximum bandwidths of the NIC of the cdm.large CDM instance is 0.8 Gbit/s and 3 Gbit/s, respectively. The theoretical maximum volume of data that can be transmitted per instance per day is about 8 TB. Similarly, the baseline and maximum bandwidths of the NIC of the cdm.xlarge instance are 4 Gbit/s and 10 Gbit/s, respectively, and the theoretical maximum volume of data that can be transmitted per instance per day is about 40 TB. The baseline and maximum bandwidths of the NIC of the cdm.4xlarge instance is 36 Gbit/s and 40 Gbit/s, respectively, and the theoretical maximum volume of data that can be transmitted per instance per day is about 360 TB. You can use multiple CDM instances if you want faster data transfer.

   The actual amount of data that can be migrated in a day depends on the data source type, the read and write performance of the source and destination, and the actual available bandwidth. Typically you can migrate as much as 8 TB per day (large file migration to OBS) using the cdm.large instance. It is recommended that you test the speed with a small amount of data before migration.
8. CDM supports incremental file migration (by skipping repeated files), but does not support resumable transfer.

   For example, if three files are to be migrated and the second file fails to be migrated due to the network fault. When the migration task is started again, the first file is skipped. The second file, however, cannot be migrated from the point where the fault occurs, but can only be migrated again.
9. During file migration, a single task supports millions of files. If there are too many files in the directory to be migrated, you are advised to split the files into different directories and create multiple tasks.
10. You can export links and jobs configured on CDM to a local directory. To ensure password security, CDM does not export the link password of the corresponding data source. Therefore, before importing job configurations to CDM, you need to manually input the password in the exported JSON file or configure the password in the import dialog box.
11. The cluster cannot automatically upgrade to a new version. You need to use the job export and import functions to upgrade the cluster to the new version.

12. If OBS is unavailable, CDM does not automatically back up users' job configurations. You need to export and back up configuration data using the export function.
13. If VPC peering connection is configured, the peer VPC subnet may overlap with the CDM management network. As a result, data sources in the peer VPC cannot be accessed. You are advised to use the public network for cross-VPC data migration, or contact the administrator to add specific routes to the VPC peering connection in the CDM background.
14. If the destination of a CDM job is a DWS or NewSQL database, constraints of the source end, such as the primary key and unique index, cannot be migrated together.
15. When performing a CDM job, ensure that the JSON file formats of the two clusters are the same so that jobs can be imported from the source cluster to the destination cluster.
16. If a running job is interrupted unexpectedly, the data that has been written to the destination will not be deleted. You must manually delete the data if needed.
17. The size of a file to be transferred cannot exceed 1 TB.

General Notes and Constraints on Database Migration

1. CDM is mainly used for batch migration. It supports only limited incremental migration but does not support real-time incremental migration. You are advised to use Data Replication Service (DRS) to migrate the incremental data of the database to RDS.
2. The entire DB migration of CDM supports only data table migration but not migration of database objects such as stored procedures, triggers, functions, and views.

   CDM applies only to scenarios where databases are migrated to the cloud at a time, including homogeneous and heterogeneous database migrations. CDM is not applicable to data synchronization, for example, disaster recovery and real-time synchronization.
3. If CDM fails to migrate an entire database or table, the data that has been imported to the target table will not be rolled back automatically. If you want to perform migration in transaction mode, configure the **Import to Staging Table** parameter to enable a rollback upon a migration failure.

   In extreme cases, the created stage table or temporary table cannot be automatically deleted. You need to manually clear the table (the name of the stage table ends with **_cdm_stage**), for example, **cdmtet_cdm_stage**).
4. If CDM needs to access data sources in the on-premises data center (for example, the on-premises MySQL database), the data sources must support Internet access and the CDM instances must be bound with elastic IP addresses. In this case, the security practice is to configure the firewall or security policies to allow only the EIPs of the CDM instances to access the local data sources.
5. Only common data types are supported, including character strings, digits, and dates. Object types are limited. If objects are too large, migration cannot be performed.
6. Only the GBK and UTF-8 character sets are supported.

7. A field name cannot contain & or %.
8. jdbc2hive and hive2jdbc entire DB migration is implemented by field name mapping, and is unavailable if the source and destination field names are inconsistent.

Permissions Configuration for Relational Database Migration

Common minimum permissions required by relational database migration:

- MySQL: You need to have the read permission on the **INFORMATION_SCHEMA** database and data tables.
- Oracle: You need to have the **resource** role and have the **select** permissions on the data table in the tablespace.
- Dameng: You need to have the **select any table** permission in the schema.
- DWS: You need to have the **schema usage** permission and the query permission on the data tables.
- SQL Server: You need to have the **sysadmin** permission.
- PostgreSQL: You need to have the **select** permission on schema tables in the database.

Constraints on FusionInsight HD and Apache Hadoop

If the FusionInsight HD and Apache Hadoop data sources are deployed in the on-premises data center, CDM must access all nodes in the cluster for reading and writing the Hadoop files. Therefore, the network access must be enabled for each node.

You are advised to use **Direct Connect** to improve the migration speed while ensuring network access.

Constraints on GaussDB(DWS)

1. If the DWS primary key or table contains only one field, the field type must be a common character string, value, or date. When data is migrated from another database to DWS, if automatic table creation is selected, the primary key must be of the following types. If no primary key is set, at least one of the following fields must be set. Otherwise, the table cannot be created and the CDM job fails.
   - INTEGER TYPES: TINYINT, SMALLINT, INT, BIGINT, NUMERIC/DECIMAL
   - CHARACTER TYPES: CHAR, BPCHAR, VARCHAR, VARCHAR2, NVARCHAR2, TEXT
   - DATA/TIME TYPES: DATE, TIME, TIMETZ, TIMESTAMP, TIMESTAMPTZ, INTERVAL, SMALLDATETIME

   NOTE

   For clusters of version 2.9.1.200 or earlier, the NVARCHAR2 data type is not supported for DWS.
2. In DWS, the character string '' is null. A null character string cannot be inserted into a field with non-null constraints. This is inconsistent with the MySQL behavior. MySQL does not consider that '' is null. Migration from MySQL to DWS may fail due to the preceding reason.

3. When the Gauss Data Service (GDS) mode is used to quickly import data to DWS, you need to configure a security group or firewall policy to allow DataNodes of DWS or FusionInsight LibrA to access port 25000 of the CDM IP address.
4. When data is imported to DWS in GDS mode, CDM automatically creates a foreign table for data import. The table name ends with a universally unique identifier (UUID), for example, **cdmtest_aecf3f8n0z73dsl72d0d1dk4lcir8cd**. If a job fails, it will be automatically deleted. In extreme cases, you may need to manually delete it.

Constraints on OBS

1. During file migration, the system automatically transfers the files concurrently. In this case, **Concurrent Extractors** in the task configuration is invalid.
2. Resumable transmission is not supported. If CDM fails to transfer files, OBS fragments are generated. You need to clear fragments on the OBS console to prevent space occupation.
3. CDM does not support the versioning control function of OBS.
4. During incremental migration, the number of files or objects in the source directory of a single job depends on the CDM cluster flavor. A cdm.large cluster supports a maximum of 300,000 files; a cdm.medium cluster supports a maximum of 200,000 files; and a cdm.small cluster supports a maximum of 100,000 files.

   If the number of files or objects in a single directory exceeds the upper limit, split the files or objects into multiple migration jobs based on subdirectories.

Constraints on DLI

- To use CDM to migrate data to DLI, you must have the read permissions of OBS.
- If the destination is DLI, you are advised to set the number of concurrent extractors to 1. Otherwise, data may fail to be written.

Constraints on Oracle

Real-time incremental data synchronization is not supported for Oracle databases.

Constraints on DCS and Redis

1. The Redis service of the third-party cloud cannot serve as the migration source. However, the Redis set up in the on-premises data center or on the ECS can be the migration source and destination.
2. Only the hash and string data formats are supported.

Constraints on DDS and MongoDB

When you migrate MongoDB or DDS data, CDM reads the first row of the collection as an example of the field list. If the first row of data does not contain all fields of the collection, you need to manually add fields.

Constraints on CSS and Elasticsearch

1. CDM supports automatic creation of indexes and field types. The index and field type names can contain only lowercase letters.
2. You cannot modify the field type under an index after it is created, but only create another field.

   If you need to modify the field type, you need to create an index or run the Elasticsearch command on Kibana to delete the existing index and create another index (the data is also deleted).
3. When the field type of the index created by CDM is date, the data format must be *yyyy-MM-dd HH:mm:ss.SSS Z*. For example, **2018-08-08 08:08:08.888 +08:00**.

   During data migration to CSS, if the original data of the **date** field does not meet the format requirements, you can use the **field conversion** function of CDM to convert the data to the preceding format.

Constraints on DIS and Kafka

- The data in the message body is a record in CSV format that supports multiple delimiters. Messages cannot be parsed in binary or other formats.
- If a job is set to run for a long time, the job will fail if the DIS system is interrupted.
- If the source is MRS Kafka, custom fields are not supported in field mapping.
- If the source is DMS Kafka, custom fields are supported in field mapping.

Constraints on CloudTable and HBase

1. When you migrate data from CloudTable or HBase, CDM reads the first row of the table as an example of the field list. If the first row of data does not contain all fields of the table, you need to manually add fields.
2. Because HBase is schema-less, CDM cannot obtain the data types. If the data is stored in binary format, CDM cannot parse the data.

Constraints on Hive

- If Hive stores timestamp data in Parquet format, timestamps are accurate to the nanosecond, for example, 2023-03-27 00:00:00.000. If the source data precision is higher than the nanosecond, the data will be truncated during field mapping. For example, if the source data is **2023-03-27 00:00:00.12345**, it will be truncated to **2023-03-27 00:00:00.123** at the destination.
- If Hive serves as the migration destination and the storage format is Textfile, delimiters must be explicitly specified in the statement for creating Hive tables. The following is an example:

```
CREATE TABLE csv_tbl(
smallint_value smallint,
tinyint_value tinyint,
int_value int,
bigint_value bigint,
float_value float,
double_value double,
decimal_value decimal(9, 7),
timestmamp_value timestamp,
date_value date,
varchar_value varchar(100),
```



```
string_value string,
char_value char(20),
boolean_value boolean,
binary_value binary,
varchar_null varchar(100),
string_null string,
char_null char(20),
int_null int
)
ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde'
WITH SERDEPROPERTIES (
"separatorChar" = "\t",
"quoteChar"     = "'",
"escapeChar"    = "\\"
)
STORED AS TEXTFILE;
```

4.3 Supported Data Sources

4.3.1 Supported Data Sources (2.9.3.300)

CDM provides the following migration modes which support different data sources:

- **Table/File migration** in the import of data into a data lake or migration of data to the cloud. For details, see **Data Sources Supported by Table/File Migration**.
- **Entire DB migration** in the import of data into a data lake or migration of data to the cloud. For details, see **Supported Data Sources in Entire DB Migration**.

NOTE

This section describes the data sources supported by CDM clusters of version 2.9.3.300. The supported data sources vary depending on the CDM cluster version.

Data Sources Supported by Table/File Migration

Table/File migration can migrate data in tables or files.

Table 4-1 describes the supported data sources.

Table 4-1 Supported data sources during table/file migration

Category	Source	Destination	Description
Data warehouse	Data Warehouse Service	• Data warehouse: GaussDB(DWS), Data Lake Insight (DLI), and MRS ClickHouse • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object-based storage: Object Storage Service (OBS) • Relational database: RDS for MySQL, RDS for PostgreSQL, RDS for SQL Server, MySQL, PostgreSQL, Microsoft SQL Server, and Oracle • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	The DWS physical machine management mode is not supported.
	Data Lake Insight (DLI)	• Data warehouse: GaussDB(DWS), Data Lake Insight (DLI), and MRS ClickHouse • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object-based storage: Object Storage Service (OBS) • Relational database: RDS for MySQL, RDS for PostgreSQL, RDS for SQL Server, MySQL, PostgreSQL, Microsoft SQL Server, and Oracle • NoSQL: CloudTable and MongoDB • Search: Elasticsearch and Cloud Search Service (CSS)	Recommended MongoDB version: 4.2
	MRS ClickHouse	Data warehouse: MRS ClickHouse and Data Lake Insight (DLI)	Recommended MRS ClickHouse version: 21.3.4.*X*

<table>
<tr><th>Category</th><th>Source</th><th>Destination</th><th>Description</th></tr>
<tr><td rowspan="3">Hadoop</td><td>MRS HDFS</td><td rowspan="2">• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI)
• Hadoop: MRS HDFS, MRS HBase, and MRS Hive
• Object-based storage: Object Storage Service (OBS)
• Relational database: RDS for MySQL, RDS for PostgreSQL, RDS for SQL Server, MySQL, PostgreSQL, Microsoft SQL Server, and Oracle
• NoSQL: CloudTable
• Search: Elasticsearch and Cloud Search Service (CSS)</td><td rowspan="3">• Supported by local storage. Only MRS Hive and MRS Hudi are supported in storage-compute decoupling scenarios.
• Only MRS Hive is supported in Ranger scenarios.
• Not supported if SSL is enabled for ZooKeeper
• Recommended MRS HDFS versions:
– 2.8.X
– 3.1.X
• Recommended MRS HBase versions:
– 2.1.X
– 1.3.X
• MRS Hive and MRS Hudi 2.x versions are not supported. The following versions are recommended:
– 1.2.X
– 3.1.X</td></tr>
<tr><td>MRS HBase</td></tr>
<tr><td>MRS Hive</td><td>• Data warehouse: GaussDB(DWS), Data Lake Insight (DLI), and MRS ClickHouse
• Hadoop: MRS HDFS, MRS HBase, and MRS Hive
• Object-based storage: Object Storage Service (OBS)
• Relational database: RDS for MySQL, RDS for PostgreSQL, RDS for SQL Server, MySQL, PostgreSQL, Microsoft SQL Server, and Oracle
• NoSQL: CloudTable
• Search: Elasticsearch and Cloud Search Service (CSS)</td></tr>
<tr><td>MRS Hudi</td><td>• Data warehouse: GaussDB(DWS)
• Hadoop: MRS HBase</td></tr>
</table>

Category	Source	Destination	Description
	FusionInsight HDFS FusionInsight HBase FusionInsight Hive	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object-based storage: Object Storage Service (OBS) • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	• FusionInsight cannot serve as the destination. • Supported only by local storage and not in storage-compute decoupling scenarios • Not supported by Ranger • Not supported if SSL is enabled for ZooKeeper • Recommended FusionInsight HDFS versions: – 2.8.X – 3.1.X • Recommended FusionInsight HBase versions: – 2.1.X – 1.3.X • Recommended FusionInsight Hive versions: – 1.2.X – 3.1.X

Category	Source	Destination	Description
	Apache HBase Apache Hive Apache HDFS	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object-based storage: Object Storage Service (OBS) • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	• Apache cannot serve as the destination. • Supported only by local storage and not in storage-compute decoupling scenarios • Not supported by Ranger • Not supported if SSL is enabled for ZooKeeper • Recommended Apache HBase versions: – 2.1.X – 1.3.X • Apache Hive 2.*x* versions are not supported. The following versions are recommended: – 1.2.X – 3.1.X • Recommended Apache HDFS versions: – 2.8.X – 3.1.X
Object storage	Object Storage Service (OBS)	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	• Object Storage Migration Service (OMS) is recommended for migration between object storage services. • Binary files cannot be imported to a database or NoSQL.

Category	Source	Destination	Description
File system	FTP SFTP	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS) • Object-based storage: Object Storage Service (OBS)	• The file system cannot serve as the destination. • Only text files such as CSV files can be migrated from FTP or SFTP servers to search services. Binary files cannot. • Only binary files can be migrated from FTP or SFTP servers to OBS. • obsutil is recommended for migrating data from HTTP servers to OBS. For details, see **Introduction to obsutil**.
	HTTP	Hadoop: MRS HDFS	

<table>
<tr><th>Category</th><th>Source</th><th>Destination</th><th>Description</th></tr>
<tr><td rowspan="6">Relational database</td><td>RDS for MySQL</td><td>• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI)
• Hadoop: MRS HDFS, MRS HBase, MRS Hive, and MRS Hudi
• Object-based storage: Object Storage Service (OBS)
• NoSQL: CloudTable
• Relational database: RDS for MySQL, RDS for PostgreSQL, and RDS for SQL Server
• Search: Elasticsearch and Cloud Search Service (CSS)</td><td rowspan="6">• You are advised to use Data Replication Service (DRS) to migrate data between OLTP databases.
• RDS for MySQL does not support the SSL mode.
• Recommended Microsoft SQL Server version: 2005 or later
• The KingBase database and GaussDB can be connected using the PostgreSQL connector. The supported source and destination are the same as those of the PostgreSQL data source.</td></tr>
<tr><td>RDS for SQL Server</td><td rowspan="2">• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI)
• Hadoop: MRS HDFS, MRS HBase, and MRS Hive
• Object-based storage: Object Storage Service (OBS)
• NoSQL: CloudTable
• Relational database: RDS for MySQL, RDS for PostgreSQL, and RDS for SQL Server
• Search: Elasticsearch and Cloud Search Service (CSS)</td></tr>
<tr><td>RDS for PostgreSQL</td></tr>
<tr><td>MySQL</td><td rowspan="3">• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI)
• Hadoop: MRS HDFS, MRS HBase, MRS Hive, and MRS Hudi
• Object-based storage: Object Storage Service (OBS)
• NoSQL: CloudTable
• Search: Elasticsearch and Cloud Search Service (CSS)</td></tr>
<tr><td>PostgreSQL</td></tr>
<tr><td>Oracle</td></tr>
<tr><td>Microsoft SQL Server</td><td>• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI)
• Hadoop: MRS HDFS, MRS HBase, and MRS Hive
• Object-based storage: Object Storage Service (OBS)
• NoSQL: CloudTable
• Search: Elasticsearch and Cloud Search Service (CSS)</td></tr>
</table>

Category	Source	Destination	Description
	SAP HANA	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS Hive	SAP HANA data sources have the following restrictions: • SAP HANA cannot serve as the destination. • Only the 2.00.050.00.1592305219 version is supported. • Only the Generic Edition is supported. • BW/4 FOR HANA is not supported. • Only database names, table names, and column names consisting of English letters are supported. Special characters such as spaces and symbols are not allowed. • The following data types are supported: date, digit, Boolean, and character (except SHORTTEXT). Other data types such as binary are not supported. • During migration, tables cannot be automatically created at the destination.

Category	Source	Destination	Description
	Database Sharding	• Data warehouse: Data Lake Insight (DLI) • Hadoop: MRS HBase and MRS Hive • Search: Elasticsearch and Cloud Search Service (CSS) • Object-based storage: Object Storage Service (OBS)	Database shards cannot serve as the destination.
	ShenTong	• Hadoop: MRS Hive and MRS Hudi	-
NoSQL	Distributed Cache Service (DCS)	Hadoop: MRS HDFS, MRS HBase, and MRS Hive	NoSQL except CloudTable cannot serve as the destination. For how to migrate data from Redis to DCS, see **Migrating Data from Self-Hosted Redis to DCS**.
	Redis		
	Document Database Service		
	MongoDB		
	CloudTable HBase	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object-based storage: Object Storage Service (OBS) • Relational database: RDS for MySQL, RDS for PostgreSQL, RDS for SQL Server, MySQL, PostgreSQL, Microsoft SQL Server, and Oracle • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	

Category	Source	Destination	Description
	Cassandra	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object-based storage: Object Storage Service (OBS) • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	
Message system	Data Ingestion Service (DIS)	Search: Cloud Search Service (CSS)	The message system cannot serve as the destination.
	Apache Kafka		
	DMS Kafka		
	MRS Kafka	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object-based storage: Object Storage Service (OBS) • Relational database: RDS for MySQL, RDS for PostgreSQL, and RDS for SQL Server • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	• MRS Kafka cannot serve as the destination. • Supported only by local storage and not in storage-compute decoupling scenarios • Not supported by Ranger • Not supported if SSL is enabled for ZooKeeper
Search	Elasticsearch	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object-based storage: Object Storage Service (OBS) • Relational database: RDS for MySQL, RDS for PostgreSQL, and RDS for SQL Server • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	Only the non-security mode is supported.
	Cloud Search Service (CSS)		You are advised to use Logstash to import data to CSS. For details, see **Using Logstash to Import Data to Elasticsearch**

NOTE

In the preceding table, the non-cloud data sources, such as MySQL, include on-premises MySQL, MySQL built on ECSs, or MySQL on the third-party cloud.

Supported Data Sources in Entire DB Migration

Entire DB migration is used when an on-premises data center or a database created on an ECS needs to be synchronized to a database service or big data service on the cloud. It is suitable for offline database migration but not online real-time migration.

Table 4-2 lists the data sources supporting entire DB migration using CDM.

Table 4-2 Supported data sources in entire DB migration

Category	Data Source	Read	Write	Description
Data warehouse	Data Warehouse Service	Supported	Supported	-
Hadoop (available only for local storage, and not for storage-compute decoupling, Ranger, or ZooKeeper for which SSL is enabled)	MRS HBase	Supported	Supported	Entire DB migration only to MRS HBase Recommended versions: • 2.1.X • 1.3.X
	MRS Hive	Supported	Supported	Entire DB migration only to a relational database 2.*x* versions are not supported. The following versions are recommended: • 1.2.X • 3.1.X
	FusionInsight HBase	Supported	Not supported	Recommended versions: • 2.1.X • 1.3.X

Category	Data Source	Read	Write	Description
	FusionInsight Hive	Supported	Not supported	Entire DB migration only to a relational database 2.*x* versions are not supported. The following versions are recommended: • 1.2.X • 3.1.X
	Apache HBase	Supported	Not supported	Recommended versions: • 2.1.X • 1.3.X
	Apache Hive	Supported	Not supported	Entire DB migration only to a relational database 2.*x* versions are not supported. The following versions are recommended: • 1.2.X • 3.1.X
	MRS Hudi	Supported	Supported	Supported only by local storage and in storage-compute decoupling scenarios 2.*x* versions are not supported. The following versions are recommended: • 1.2.X • 3.1.X

Category	Data Source	Read	Write	Description
Relational database	RDS for MySQL	Supported	Supported	Migration from OLTP to OLTP is not supported. In this scenario, you are advised to use the Data Replication Service (DRS).
	RDS for PostgreSQL	Supported	Supported	
	RDS for SQL Server	Supported	Supported	
	MySQL	Supported	Not supported	
	PostgreSQL	Supported	Not supported	
	Microsoft SQL Server	Supported	Not supported	
	Oracle	Supported	Not supported	

Category	Data Source	Read	Write	Description
	SAP HANA	Supported	Not supported	• Only the 2.00.050.00.1592305219 version is supported. • Only the Generic Edition is supported. • BW/4 FOR HANA is not supported. • Only database names, table names, and column names consisting of English letters are supported. Special characters such as spaces and symbols are not allowed. • The following data types are supported: date, digit, Boolean, and character (except SHORTTEXT). Other data types such as binary are not supported. • During migration, tables cannot be automatically created at the destination.
	Dameng database	Supported	Not supported	Only to DWS and Hive

Category	Data Source	Read	Write	Description
NoSQL	Distributed Cache Service (DCS)	Not supported	Supported	Only migration from MRS to DCS is supported.
	Document Database Service (DDS)	Supported	Supported	Only migration between DDS and MRS is supported.
	CloudTable	Supported	Supported	-

4.3.2 Supported Data Sources (2.9.2.200)

CDM provides the following migration modes which support different data sources:

- **Table/File migration** in the import of data into a data lake or migration of data to the cloud. For details, see **Data Sources Supported by Table/File Migration**.
- **Entire DB migration** in the import of data into a data lake or migration of data to the cloud. For details, see **Supported Data Sources in Entire DB Migration**.

NOTE

This section describes the data sources supported by CDM clusters of version 2.9.2.200. The supported data sources vary depending on the CDM cluster version.

Data Sources Supported by Table/File Migration

Table/File migration can migrate data in tables or files.

Table 4-3 describes the supported data sources.

Table 4-3 Supported data sources during table/file migration

Category	Source	Destination	Description
Data warehouse	GaussDB(DWS)	• Data warehouse: GaussDB(DWS), Data Lake Insight (DLI), and MRS ClickHouse • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object storage: Object Storage Service (OBS) • Relational database: RDS for MySQL, RDS for PostgreSQL, RDS for SQL Server, MySQL, PostgreSQL, Microsoft SQL Server, and Oracle • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	The DWS physical machine management mode is not supported.
	Data Lake Insight (DLI)		-
	MRS ClickHouse	Data warehouse: MRS ClickHouse and Data Lake Insight (DLI)	Recommended MRS ClickHouse version: 21.3.4.*X*

<table>
<tr><th>Category</th><th>Source</th><th>Destination</th><th>Description</th></tr>
<tr><td rowspan="3">Hadoop</td><td>MRS HDFS
MRS HBase</td><td>• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI)
• Hadoop: MRS HDFS, MRS HBase, and MRS Hive
• Object storage: Object Storage Service (OBS)
• Relational database: RDS for MySQL, RDS for PostgreSQL, RDS for SQL Server, MySQL, PostgreSQL, Microsoft SQL Server, and Oracle
• NoSQL: CloudTable
• Search: Elasticsearch and Cloud Search Service (CSS)</td><td rowspan="3">• Supported by local storage. Only MRS Hive and MRS Hudi are supported in storage-compute decoupling scenarios.
• Only MRS Hive is supported in Ranger scenarios.
• Not supported if SSL is enabled for ZooKeeper
• Recommended MRS HDFS versions:
– 2.8.X
– 3.1.X
• Recommended MRS HBase versions:
– 2.1.X
– 1.3.X
• MRS Hive and MRS Hudi 2.x versions are not supported. The following versions are recommended:
– 1.2.X
– 3.1.X</td></tr>
<tr><td>MRS Hive</td><td>• Data warehouse: GaussDB(DWS), Data Lake Insight (DLI), and MRS ClickHouse
• Hadoop: MRS HDFS, MRS HBase, and MRS Hive
• Object storage: Object Storage Service (OBS)
• Relational database: RDS for MySQL, RDS for PostgreSQL, RDS for SQL Server, MySQL, PostgreSQL, Microsoft SQL Server, and Oracle
• NoSQL: CloudTable
• Search: Elasticsearch and Cloud Search Service (CSS)</td></tr>
<tr><td>MRS Hudi</td><td>Data warehouse: GaussDB(DWS)</td></tr>
</table>

Category	Source	Destination	Description
	FusionInsight HDFS FusionInsight HBase FusionInsight Hive	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object storage: Object Storage Service (OBS) • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	• FusionInsight cannot serve as the destination. • Supported only by local storage and not in storage-compute decoupling scenarios • Not supported by Ranger • Not supported if SSL is enabled for ZooKeeper • Recommended FusionInsight HDFS versions: – 2.8.X – 3.1.X • Recommended FusionInsight HBase versions: – 2.1.X – 1.3.X • Recommended FusionInsight Hive versions: – 1.2.X – 3.1.X

Category	Source	Destination	Description
	Apache HBase Apache Hive Apache HDFS	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object storage: Object Storage Service (OBS) • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	• Apache cannot serve as the destination. • Supported only by local storage and not in storage-compute decoupling scenarios • Not supported by Ranger • Not supported if SSL is enabled for ZooKeeper • Recommended Apache HBase versions: – 2.1.X – 1.3.X • Apache Hive 2.*x* versions are not supported. The following versions are recommended: – 1.2.X – 3.1.X • Recommended Apache HDFS versions: – 2.8.X – 3.1.X
Object storage	Object Storage Service (OBS)	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	• Object Storage Migration Service (OMS) is recommended for migration between object storage services. • Binary files cannot be imported to a database or NoSQL.

Category	Source	Destination	Description
File system	FTP SFTP	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	• The file system cannot serve as the destination. • Only text files such as CSV files can be migrated from FTP or SFTP servers to search services. Binary files cannot. • obsutil is recommended for migrating data from HTTP servers to OBS. For details, see **Introduction to obsutil**.
	HTTP	Hadoop: MRS HDFS	
Relational database	RDS for MySQL	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, MRS Hive, and MRS Hudi • Object storage: Object Storage Service (OBS) • NoSQL: CloudTable • Relational database: RDS for MySQL, RDS for PostgreSQL, and RDS for SQL Server • Search: Elasticsearch and Cloud Search Service (CSS)	• You are advised to use Data Replication Service (DRS) to migrate data between OLTP databases. • RDS for MySQL does not support the SSL mode. • Recommended Microsoft SQL Server version: 2005 or later • The KingBase database and GaussDB can be connected using the PostgreSQL connector. The supported source and destination are the same as those of the PostgreSQL data source.
	RDS for SQL Server RDS for PostgreSQL	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object storage: Object Storage Service (OBS) • NoSQL: CloudTable • Relational database: RDS for MySQL, RDS for PostgreSQL, and RDS for SQL Server • Search: Elasticsearch and Cloud Search Service (CSS)	

Category	Source	Destination	Description
	MySQL PostgreSQL Oracle	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, MRS Hive, and MRS Hudi • Object-based storage: Object Storage Service (OBS) • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	
	Microsoft SQL Server	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object-based storage: Object Storage Service (OBS) • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	

Category	Source	Destination	Description
	SAP HANA	• Data warehouse: Data Lake Insight (DLI) • Hadoop: MRS Hive	SAP HANA data sources have the following restrictions: • SAP HANA cannot serve as the destination. • Only the 2.00.050.00.1592305219 version is supported. • Only the Generic Edition is supported. • BW/4 FOR HANA is not supported. • Only database names, table names, and column names consisting of English letters are supported. Special characters such as spaces and symbols are not allowed. • The following data types are supported: date, digit, Boolean, and character (except SHORTTEXT). Other data types such as binary are not supported. • During migration, tables cannot be automatically created at the destination.

Category	Source	Destination	Description
	Database sharding	• Data warehouse: Data Lake Insight (DLI) • Hadoop: MRS HBase and MRS Hive • Search: Elasticsearch and Cloud Search Service (CSS) • Object-based storage: Object Storage Service (OBS)	Database shards cannot serve as the destination. A shard link connects to multiple backend data sources at the same time. The link can be used as the job source to migrate data from those data sources to other data sources.
NoSQL	Redis	Hadoop: MRS HDFS, MRS HBase, and MRS Hive	NoSQL except CloudTable cannot serve as the destination.
	Document Database Service (DDS)		
	MongoDB		
	CloudTable HBase	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object storage: Object Storage Service (OBS) • Relational database: RDS for MySQL, RDS for PostgreSQL, RDS for SQL Server, MySQL, PostgreSQL, Microsoft SQL Server, and Oracle • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	
	Cassandra	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object storage: Object Storage Service (OBS) • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	

Category	Source	Destination	Description
Message system	Data Ingestion Service (DIS)	Search: Cloud Search Service (CSS)	The message system cannot serve as the destination.
	Apache Kafka		
	DMS Kafka		
	MRS Kafka	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object storage: Object Storage Service (OBS) • Relational database: RDS for MySQL, RDS for PostgreSQL, and RDS for SQL Server • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	• MRS Kafka cannot serve as the destination. • Supported only by local storage and not in storage-compute decoupling scenarios • Not supported by Ranger • Not supported if SSL is enabled for ZooKeeper
Search	Elasticsearch	• Data warehouse: GaussDB(DWS) and Data Lake Insight (DLI) • Hadoop: MRS HDFS, MRS HBase, and MRS Hive • Object storage: Object Storage Service (OBS) • Relational database: RDS for MySQL, RDS for PostgreSQL, and RDS for SQL Server • NoSQL: CloudTable • Search: Elasticsearch and Cloud Search Service (CSS)	Only the non-security mode is supported.
	Cloud Search Service (CSS)		You are advised to use Logstash to import data to CSS. For details, see **Using Logstash to Import Data to Elasticsearch**

NOTE

In the preceding table, the non-cloud data sources, such as MySQL, include on-premises MySQL, MySQL built on ECSs, or MySQL on the third-party cloud.

Supported Data Sources in Entire DB Migration

Entire DB migration is used when an on-premises data center or a database created on an ECS needs to be synchronized to a database service or big data

service on the cloud. It is suitable for offline database migration but not online real-time migration.

Table 4-4 lists the data sources supporting entire DB migration using CDM.

Table 4-4 Supported data sources in entire DB migration

Category	Data Source	Read	Write	Description
Data warehouse	Data Warehouse Service (DWS)	Supported	Supported	-
Hadoop (available only for local storage, and not for storage-compute decoupling, Ranger, or ZooKeeper for which SSL is enabled)	MRS HBase	Supported	Supported	Entire DB migration only to MRS HBase Recommended versions: • 2.1.X • 1.3.X
	MRS Hive	Supported	Supported	Entire DB migration only to a relational database 2.*x* versions are not supported. The following versions are recommended: • 1.2.X • 3.1.X
	FusionInsight HBase	Supported	Not supported	Recommended versions: • 2.1.X • 1.3.X
	FusionInsight Hive	Supported	Not supported	Entire DB migration only to a relational database 2.*x* versions are not supported. The following versions are recommended: • 1.2.X • 3.1.X

Category	Data Source	Read	Write	Description
	Apache HBase	Supported	Not supported	Recommended versions: • 2.1.X • 1.3.X
	Apache Hive	Supported	Not supported	Entire DB migration only to a relational database 2.*x* versions are not supported. The following versions are recommended: • 1.2.X • 3.1.X
Relational database	RDS for MySQL	Supported	Supported	Migration from OLTP to OLTP is not supported. In this scenario, you are advised to use the Data Replication Service (DRS).
	RDS for PostgreSQL	Supported	Supported	
	RDS for SQL Server	Supported	Supported	
	MySQL	Supported	Not supported	
	PostgreSQL	Supported	Not supported	
	Microsoft SQL Server	Supported	Not supported	
	Oracle	Supported	Not supported	

Category	Data Source	Read	Write	Description
	SAP HANA	Supported	Not supported	• Only the 2.00.050.00.1592305219 version is supported. • Only the Generic Edition is supported. • BW/4 FOR HANA is not supported. • Only database names, table names, and column names consisting of English letters are supported. Special characters such as spaces and symbols are not allowed. • The following data types are supported: date, digit, Boolean, and character (except SHORTTEXT). Other data types such as binary are not supported. • During migration, tables cannot be automatically created at the destination.
	Dameng database	Supported	Not supported	Only to DWS and Hive

Category	Data Source	Read	Write	Description
NoSQL	Redis	Supported	Supported	-
	Document Database Service (DDS)	Supported	Supported	Only migration between DDS and MRS is supported.
	CloudTable Service (CloudTable)	Supported	Supported	-

4.3.3 Supported Data Types

To ensure that data is completely imported to the migration destination, correctly configure field mappings based on data types supported for different data sources. For details, see **Table 4-5**.

Table 4-5 Supported data types

Data Connection Type	Data Type
MySQL	**Data Types Supported in MySQL Database Migration**
SQL Server	**Data Types Supported in SQL Server Database Migration**
Oracle	**Data Types Supported in Oracle Database Migration**
PostgreSQL	**Data Types Supported in PostgreSQL Database Migration**
ShenTong	**Data Types Supported in ShenTong Database Migration**
SAP HANA	**Data Types Supported in SAP HANA Database Migration**
DWS	**Data Types Supported in DWS Database Migration**
Dameng	**Data Types Supported in Dameng Database Migration**
DLI	**Data Types Supported in DLI Database Migration**
Elasticsearch/Cloud Search Service (CSS)	**Data Types Supported in Elasticsearch/CSS Database Migration**

Data Types Supported in MySQL Database Migration

When the source end is a MySQL database and the destination end is a Hive or DWS database, the following data types are supported:

Table 4-6 Data types supported for the open-source MySQL database

Category	Type	Description	Storage Format Example	Hive	DWS
Character string	CHAR(M)	A fixed-length string of 1 to 255 characters, for example, CHAR(5). The length limit is not mandatory. It is set to 1 by default.	'a' or 'aaaaa'	CHAR	CHAR
	VARCHAR(M)	A variable-length string of 1 to 255 characters (more than 255 characters for MySQL of a later version), for example, VARCHAR(25). When creating a field of the VARCHAR type, you must define the length.	'a' or 'aaaaa'	VARCHAR	VARCHAR
Value	DECIMAL(M, D)	Uncompressed floating-point numbers cannot be unsigned. In unpacking decimals, each decimal corresponds to a byte. Defining the number of display lengths (M) and decimals (D) is required. NUMERIC is the synonym of DECIMAL.	52.36	DECIMAL	When D is 0, it corresponds to BIGINT. When D is not 0, it corresponds to NUMBERIC.
	NUMBERIC	Same as DECIMAL	-	DECIMAL	NUMBERIC

Category	Type	Description	Storage Format Example	Hive	DWS
	INTEGER	An integer of normal size that can be signed. If the value is signed, it ranges from -2147483648 to 2147483647. If the value is unsigned, the value ranges from 0 to 4294967295. Up to 11-bit width can be specified.	5236	INT	INTEGER
	INTEGER UNSIGNED	Unsigned form of INTEGER	-	BIGINT	INTEGER
	INT	Same as INTEGER	5236	INT	INTEGER
	INT UNSIGNED	Same as INTEGER UNSIGNED	-	BIGINT	INTEGER
	BIGINT	A large integer that can be signed. If the value is signed, it ranges from -9223372036854775808 to 9223372036854775807. If the value is unsigned, the value ranges from 0 to 18446744073709551615. Up to 20-bit width can be specified.	5236	BIGINT	BIGINT
	BIGINT UNSIGNED	Unsigned form of BIGINT	-	BIGINT	BIGINT

Category	Type	Description	Storage Format Example	Hive	DWS
	MEDIUMINT	A medium-sized integer that can be signed. If the value is signed, it ranges from -8388608 to 8388607. If the value is unsigned, it ranges from 0 to 16777215, and you can specify a maximum of 9-bit width.	-128 to 127	INT	INTEGER
	MEDIUMINT UNSIGNED	Unsigned form of MEDIUMINT	-	BIGINT	INTEGER
	TINYINT	A very small integer that can be signed. If signed, the value ranges from -128 to 127. If unsigned, the value ranges from 0 to 255, and you can specify a maximum of 4-bit width.	100	TINYINT	SMALLINT
	TINYINT UNSIGNED	Unsigned form of TINYINT	-	TINYINT	SMALLINT
	BOOL	The bool of MySQL is tinyint(1).	-128, 127	SMALLINT	BYTEA
	SMALLINT	A small integer that can be signed. If the value is signed, it ranges from -32768 to 32767. If unsigned, the value ranges from 0 to 65535, and you can specify a maximum of 5-bit width.	9999	SMALLINT	SMALLINT

Category	Type	Description	Storage Format Example	Hive	DWS
	SMALLINT UNSIGNED	Unsigned form of SMALLINT	-	INT	SMALLINT
	REAL	Same as DOUBLE	-	DOUBLE	-
	FLOAT(M,D)	Unsigned floating-point numbers cannot be used. The display length (M) and number of decimal places (D) can be specified. This is not mandatory, and the default value is 10,2. In the preceding information, 2 indicates the number of decimal places and 10 indicates the total number of digits (including decimal places). The decimal precision can reach 24 floating points.	52.36	FLOAT	FLOAT4
	DOUBLE(M,D)	Unsigned double-precision floating-point numbers cannot be used. The display length (M) and number of decimal places (D) can be specified. This is not mandatory. The default value is 16,4, where 4 is the number of decimal places. The decimal precision can reach 53-digit. REAL is a synonym of DOUBLE.	52.36	DOUBLE	FLOAT8
	DOUBLE PRECISION	Similar to DOUBLE	52.3	DOUBLE	FLOAT8

Category	Type	Description	Storage Format Example	Hive	DWS
Bit	BIT(M)	Stored bit type value. BIT(M) can store up to *M* bits of values, and *M* ranges from 1 to 64.	B'1111100' B'1100'	TINYINT	BYTEA
Time and date	DATE	The value is in the *YYYY-MM-DD* format and ranges from **1000-01-01** to **9999-12-31**. For example, **December 30, 1973** will be stored as **1973-12-30**.	1999-10-01	DATE	TIMESTAMP
	TIME	Stores information about the hour, minute, and second.	'09:10:21' or '9:10:21'	Not supported (string)	TIME
	DATETIME	The date and time are in the *YYYY-MM-DD HH:MM:SS* format and range from **1000-01-01 00:00:00** to **9999-12-31 23:59:59**. For example, **3:30 p.m. on December 30, 1973** will be stored as **1973-12-30 15:30:00**.	'1973-12-30 15:30:00'	TIMESTAMP	TIMESTAMP
	TIMESTAMP	Timestamp type. Timestamp between midnight on January 1, 1970 and a time point in 2037. Similar to the DATETIME format (YYYYMMDDHHMMSS), except that no hyphen is required. For example, **3:30 p.m. December 30, 1973** will be stored as **19731230153000**.	19731230153000	TIMESTAMP	TIMESTAMP

Category	Type	Description	Storage Format Example	Hive	DWS
	YEAR(M)	The year is stored in 2-digit or 4-digit number format. If the length is specified as 2 (for example, YEAR(2)), the year ranges from 1970 to 2069 (70 to 69). If the length is specified as 4, the year ranges from 1901 to 2155. The default length is 4.	2000	Not supported (string)	Not supported
Multimedia (binary)	BINARY(M)	The number of bytes is *M*. The length of a variable-length binary string ranges from 0 to *M*. *M* is the value length plus 1.	0x2A3B4058 (binary data)	Not supported	BYTEA
	VARBINARY(M)	The number of bytes is *M*. A fixed binary string with a length of 0 to *M*.	0x2A3B4059 (binary data)	Not supported	BYTEA
	TEXT	The maximum length of the field is 65535 characters. TEXT is a "binary large object" and is used to store large binary data, such as images or other types of files.	0x5236 (binary data)	Not supported	Not supported
	TINYTEXT	A binary string of 0 to 255 bytes in short text	-	-	Not supported
	MEDIUMTEXT	A binary string of 0 to 167772154 bytes in medium-length text	-	-	Not supported
	LONGTEXT	A binary string of 0 to 4294967295 bytes in large-length text	-	-	Not supported

Category	Type	Description	Storage Format Example	Hive	DWS
	BLOB	The maximum length of the field is 65535 characters. BLOB is a "binary large object" and is used to store large binary data, such as images or other types of files. BLOB is case-sensitive.	0x5236 (binary data)	Not supported	Not supported
	TINYBLOB	A binary string of 0 to 255 bytes in short text	-	Not supported	Not supported
	MEDIUMBLOB	A binary string of 0 to 167772154 bytes in medium-length text	-	Not supported	Not supported
	LONGBLOB	A binary string of 0 to 4294967295 bytes in large-length text	0x5236 (binary data)	Not supported	Not supported
Special type	SET	SET is a string object that can have no or multiple values. The values come from the allowed column of values specified when the table is created. When specifying the SET column values that contain multiple SET members, separate the members with commas (,). The SET member value cannot contain commas (,).	-	-	Not supported
	JSON	-	-	Not supported	Not supported (TEXT)

Category	Type	Description	Storage Format Example	Hive	DWS
	ENUM	When an ENUM is defined, a list of its values is created, which are the items that must be used for selection (or NULL). For example, if you want a field to contain "A", "B", or "C", you can define an ENUM ("A", "B", or "C"). Only these values (or NULL) can be used to fill in the field.	-	Not supported	Not supported

Data Types Supported in Oracle Database Migration

When the source end is an Oracle database and the destination end is a Hive or DWS database, the following data sources are supported:

Table 4-7 Data types supported for the Oracle database

Category	Type	Description	Hive	DWS
Character string	char	Fixed-length character string, which is padded with spaces to reach the maximum length.	CHAR	CHAR
	nchar	Fixed-length character string contains data in Unicode format.	CHAR	CHAR
	varchar2	Synonym of VARCHAR. It is a variable-length string, unlike the CHAR type, which does not pad the field or variable to reach its maximum length with spaces.	VARCHAR	VARCHAR
	nvarchar2	Variable-length character string contains data in Unicode format.	VARCHAR	VARCHAR
Value	number	Stores numbers with a precision of up to 38 digits.	DECIMAL	NUMERIC

Category	Type	Description	Hive	DWS
	binary_float	2-bit single-precision floating point number	FLOAT	FLOAT8
	binary_double	64-bit double-precision floating point number	DOUBLE	FLOAT8
	long	A maximum of 2 GB character data can be stored.	Not supported	Not supported
Time and date	date	7-byte date/time data type, including seven attributes: century, year in the century, month, day in the month, hour, minute, and second.	DATE	TIMESTAMP
	timestamp	7-byte or 11-byte fixed-width date/time data type that contains decimals (seconds)	TIMESTAMP	TIMESTAMP
	timestamp with time zone	3-byte timestamp, which supports the time zone.	TIMESTAMP	TIME WITH TIME ZONE
	timestamp with local time zone	7-byte or 11-byte fixed-width date/time data type. Time zone conversion occurs when data is inserted or read.	TIMESTAMP	Not supported (TEXT)
	interval year to month	5-byte fixed-width data type, which is used to store a time segment.	Not supported	Not supported (TEXT)
	interval day to second	11-byte fixed-width data type, which is used to store a time segment. The time segment is stored in days/hours/minutes/seconds. The value can also contain nine decimal places (seconds).	Not supported	Not supported (TEXT)
Multimedia (binary)	raw	A variable-length binary data type. Character set conversion is not performed for data stored in this data type.	Not supported	Not supported

Category	Type	Description	Hive	DWS
	long raw	Stores up to 2 GB binary information.	Not supported	Not supported
	blob	A maximum of 4 GB data can be stored.	Not supported	Not supported
	clob	In Oracle 10g and later versions, a maximum of (4 GB) x (database block size) bytes of data can be stored. CLOB contains the information for which character set conversion is to be performed. This data type is ideal for storing plain text information.	String	Not supported
	nclob	This type can store a maximum of 4 GB data. When the character set is converted, this type is affected.	Not supported	Not supported
	bfile	An Oracle directory object and a file name can be stored in the database column, and the file can be read through the Oracle directory object and file name.	Not supported	Not supported
Others	rowid	It is the address of a row in the database table. It is 10 bytes long.	Not supported	Not supported
	urowid	It is a common row ID and does not have a fixed rowid table.	Not supported	Not supported

Data Types Supported in SQL Server Database Migration

When the source end is a SQL Server database and the destination end is a Hive, Oracle or DWS database, the following data sources are supported:

Table 4-8 Data types supported for the SQL Server database

Category	Type	Description	Hive	DWS	Oracle
String data type	char	Fixed-length character string, which is padded with spaces to reach the maximum length.	CHAR	CHAR	CHAR

Category	Type	Description	Hive	DWS	Oracle
	nchar	Fixed-length character string contains data in Unicode format.	CHAR	CHAR	CHAR
	varchar	A variable-length string consists of 1 to 255 characters (more than 255 characters for MySQL of a later version). Example: VARCHAR(25). When creating a field of the VARCHAR type, you must define the length.	VARCHAR	VARCHAR	VARCHAR
	nvarchar	Stores variable-length Unicode character data, similar to varchar.	VARCHAR	VARCHAR	VARCHAR
Numeric data type	int	int is stored in four bytes, where one binary bit represents a sign bit, and the other 31 binary bits represent a length and a size, and may represent all integers ranging from -2^{31} to $2^{31} - 1$.	INT	INTEGER	INT
	bigint	bigint is stored in eight bytes, where one binary bit represents a sign bit, and the other 63 binary bits represent a length and a size, and may represent all integers ranging from -2^{63} to $2^{63} - 1$.	BIGINT	BIGINT	NUMBER
	smallint	Data of the smallint type occupies two bytes of storage space. One binary bit indicates a positive or negative sign of an integer value, and the other 15 binary bits indicate a length and a size, and may represent all integers ranging from -2^{15} to 2^{15}.	SMALLINT	SMALLINT	NUMBER
	tinyint	Tinyint data occupies one byte of storage space and can represent all integers ranging from 0 to 255.	TINYINT	TINYINT	NUMBER
	real	The value can be a positive or negative decimal number.	DOUBLE	FLOAT4	NUMBER
	float	The number of digits (in scientific notation) of the mantissa of a float value, which determines the precision and storage size	FLOAT	FLOAT8	binary_float
	decimal	Numeric data type with fixed precision and scale	DECIMAL	NUMERIC	NUMBER

Category	Type	Description	Hive	DWS	Oracle
	numeric	Stores zero, positive, and negative fixed point numbers.	DECIMAL	NUMERIC	NUMBER
Date and time data type	date	Stores date data represented by strings.	DATE	TIMESTAMP	DATE
	time	Time of a day, which is recorded in the form of a character string.	Not supported (string)	TIME	Not supported
	datetime	Stores time and date data.	TIMESTAMP	TIMESTAMP	Not supported
	datetime2	Extended type of datetime, which has a larger data range. By default, the minimum precision is the highest, and the user-defined precision is optional.	TIMESTAMP	TIMESTAMP	Not supported
	smalldatetime	The smalldatetime type is similar to the datetime type. The difference is that the smalldatetime type stores data from January 1, 1900 to June 6, 2079. When the date and time precision is low, the smalldatetime type can be used. Data of this type occupies 4-byte storage space.	TIMESTAMP	TIMESTAMP	Not supported
	datetimeoffset	A time that uses the 24-hour clock and combined with date and the time zone.	Not supported (string)	TIMESTAMP	Not supported
Multimedia data types (binary)	text	Stores text data.	Not supported (string)	Not supported (string)	Not supported
	netxt	The function of this type is the same as that of the text type. It is non-Unicode data with variable length.	Not supported (string)	Not supported (string)	Not supported

Category	Type	Description	Hive	DWS	Oracle
	image	Variable-length binary data used to store pictures, catalog pictures, or paintings.	Not supported (string)	Not supported (string)	Not supported
	binary	Binary data with a fixed length of *n* bytes, where *n* ranges from 1 to 8,000.	Not supported (string)	Not supported (string)	Not supported
	varbinary	Variable-length binary data	Not supported (string)	Not supported (string)	Not supported
Currency data type	money	Stores currency values.	Not supported (string)	Not supported (string)	Not supported
	small money	Similar to the money type, a currency symbol is prefixed to the input data. For example, the currency symbol of CNY is ¥.	Not supported (string)	Not supported (string)	Not supported
Data type	bit	Bit data type. The value is 0 or 1. The length is 1 byte. A bit value is often used as a logical value to determine whether it is true(1) or false(0). If a non-zero value is entered, the system replaces it with 1.	Not supported	Not supported	Not supported
Other data types	rowversion	Each piece of data has a counter. The value of the counter increases when an insert or update operation is performed on a table that contains the **rowversion** column in the database.	Not supported	Not supported	Not supported

Category	Type	Description	Hive	DWS	Oracle
	unique identifier	A 16-byte globally unique identifier (GUID) is a unique number generated by the SQL Server based on the network adapter address and host CPU clock. Each GUID is a hexadecimal number ranging from 0 to 9 or a to f.	Not supported	Not supported	Not supported
	cursor	Cursor data type	Not supported	Not supported	Not supported
	sql_variant	Stores any valid SQL Server data except the text, image, and timestamp data, which facilitates the development of the SQL Server.	Not supported	Not supported	Not supported
	table	Stores the result set after a table or view is processed.	Not supported	Not supported	Not supported
	xml	Data type of the XML data. XML instances can be stored in columns or variables of the XML type. The stored XML instance size cannot exceed 2 GB.	Not supported	Not supported	Not supported

Data Types Supported in PostgreSQL Database Migration

When the source end is a PostgreSQL database and the destination end is Hive, DLI, or DWS, the following data types are supported:

Table 4-9 Data types supported for the PostgreSQL database

Category	Type	Description	Hive	DWS	DLI
Character	char	Fixed-length string, which is padded to a specified length with spaces on the right.	CHAR	CHAR	Not supported (string)

Category	Type	Description	Hive	DWS	DLI
	varchar	Variable-length string. Fields or variables are not padded to the maximum length with spaces.	CARCHAR	CARCHAR	Not supported (string)
Value	smallint	The extension name int2 is stored in two bytes and ranges from –32768 to 32767.	SMALLINT	SMALLINT	SMALLINT
	int	The extension name int4 is stored in four bytes and ranges from –2147483648 to 2147483647.	INTEGER	INT	INT
	bigint	The extension name int8 is stored in eight bytes and ranges from –9223372036854775808 to 9223372036854775807.	BIGINT	BIGINT	BIGINT
	decimal(p,s)	The precision p represents the number of valid digits stored in the value, and the scale s represents the number of digits after the decimal point that can be stored. The maximum value of p is 1000.	DECIMAL(P,S)	DECIMAL(P,S)	DECIMAL(P,S)

Category	Type	Description	Hive	DWS	DLI
	float	4-byte or 8-byte storage. float(n): For the single precision, the value of n ranges from 1 to 24, the number of valid precision digits is 6, and the length is four bytes. For the double precision, the value of n ranges from 25 to 53, the number of valid precision digits is 15, and the length is 8 bytes.	FLOAT/ DOUBLE	FLOAT/ DOUBLE	FLOAT/ DOUBLE
	smallserial	Sequence data type, which is stored in smallint format	SMALLINT	SMALLINT	SMALLINT
	serial	Sequence data type, which is stored in int format	INTEGER	INT	INT
	bigserial	Sequence data type, which is stored in bigint format	BIGINT	BIGINT	BIGINT
Time and date	date	Stores the date.	DATE	DATE	DATE
	timestamp	Stores date and time data without time zones.	TIMESTAMP	TIMESTAMP	Not supported (string)
	timestamptz	Stores the date and time, including the time zone.	TIMESTAMP	TIMESTAMPZ	Not supported (string)
	time	Time within one day, excluding the time zone	Not supported (string)	TIME	Not supported (string)
	timez	Time within one day, including the time zone	Not supported (string)	TIMEZ	Not supported (string)

Category	Type	Description	Hive	DWS	DLI
	interval	Time interval	Not supported (string)	Not supported (string)	Not supported (string)
Bit string	bit	Fixed-length string, for example, **b'000101'**	Not supported (string)	Not supported (string)	Not supported (string)
	varbit	Variable-length string, for example, **b'101'**	Not supported (string)	Not supported (string)	Not supported (string)
Currency type	money	The value is stored in eight bytes and ranges from –922337203685477.5808 to 922337203685477.5807.	DOUBLE	MONEY	DECIMAL(P,S)
Boolean	boolean	The value is stored in one byte and can be **1**, **0**, or **NULL**.	BOOLEAN	BOOLEAN	BOOLEAN
Text type	text	Variable-length text without a length limit	Not supported (string)	Not supported (string)	Not supported (string)

Data Types Supported in DWS Database Migration

If the migration source is a DWS database, the following data types are supported.

Table 4-10 Data types supported for the DWS database

Category	Type	Description
Character	char	Fixed-length string, which is padded to a specified length with spaces on the right.
	varchar	Variable-length string. Fields or variables are not padded to the maximum length with spaces.
Value	double	Stores double-precision floating-point numbers.

Category	Type	Description
	decimal(p,s)	The precision p represents the number of valid digits stored in the value, and the scale s represents the number of digits after the decimal point that can be stored. The maximum value of p is 1000.
	numeric	Stores zero, positive, and negative fixed point numbers.
	real	Same as double
	int	int is stored in four bytes, where one binary bit represents a sign bit, and the other 31 binary bits represent a length and a size, and may represent all integers ranging from -2^{31} to $2^{31} - 1$.
	bigint	bigint is stored in eight bytes, where one binary bit represents a sign bit, and the other 63 binary bits represent a length and a size, and may represent all integers ranging from -2^{63} to $2^{63} - 1$.
	smallint	Data of the smallint type occupies two bytes of storage space. One binary bit indicates a positive or negative sign of an integer value, and the other 15 binary bits indicate a length and a size, and may represent all integers ranging from -2^{15} to 2^{15}.
	tinyint	Tinyint data occupies one byte of storage space and can represent all integers ranging from 0 to 255.
Time and date	date	Stores the date.
	timestamp	Stores date and time data without time zones.
	time	Time within one day, excluding the time zone
Bit string	bit	Fixed-length string, for example, **b'000101'**
Boolean	boolean	The value is stored in one byte and can be **1**, **0**, or **NULL**.
Text type	text	Variable-length text without a length limit

Data Types Supported in ShenTong Database Migration

When the source is a ShenTong database and the destination is MRS Hive or MRS Hudi, the following data types are supported.

Table 4-11 Data types supported for the ShenTong database

Category	Type	Description	Storage Format Example	MRS Hive	MRS Hudi
Character	VARCHAR	Stores specified fixed-length character strings.	'a' or 'aaaaa'	VARCHAR(765)	STRING
	BPCHAR	Stores specified variable-length character strings.	'a' or 'aaaaa'	VARCHAR(765)	STRING
Value	NUMERIC	Stores zero, positive, and negative fixed point numbers.	52.36	DECIMAL(10, 0)	DECIMAL(18, 0)
	INT	Stores zero, positive, and negative fixed point numbers.	5236	INT	INT
	BIGINT	Stores signed integers. Integer part: 19 digits; decimal part: 0 digits	5236	BIGINT	BIGINT
	TINYINT	Stores signed integers. Integer part: 3 digits; decimal part: 0 digits	100	SMALLINT	INT
	BINARY	Stores fixed-length binary data.	0x2A3B4058	Not supported	FLOAT
	VARBINARY	Stores variable-length binary data.	0x2A3B4058	Not supported	BINARY
	FLOAT	Stores floating-point numbers with binary precision.	52.36	FLOAT	FLOAT
	DOUBLE	Stores double-precision floating-point numbers.	52.3	DOUBLE	DOUBLE

Category	Type	Description	Storage Format Example	MRS Hive	MRS Hudi
Time and date	DATE	Stores information about the year, month, and day.	'1999-10-01', '1999/10/01', or '1999.10.01'	DATE	DATE
	TIME	Stores information about the hour, minute, and second.	'09:10:21' or '9:10:21'	STRING	STRING
	TIMESTAMP	Stores information about the year, month, day, hour, minute, and second.	'2002-12-12 09:10:21', '2002-12-12 9:10:21', '2002/12/12 09:10:21', or '2002.12.12 09:10:21'	TIMESTAMP	TIMESTAMP
Multimedia	CLOB	Stores variable-length binary large objects with a maximum length of 2 GB minus 1 byte.	0x5236 (binary data)	STRING	STRING
	BLOB	Stores variable-length binary large objects with a maximum length of 2 GB minus 1 byte.	0x5236 (binary data)	Not supported	BINARY
Boolean	BOOLEAN	The value is stored in one byte and can be **1**, **0**, or **NULL**.	1	BOOLEAN	BOOLEAN

Data Types Supported in SAP HANA Database Migration

If the source is an SAP HANA database, the following data types are supported.

Table 4-12 Data types supported for the SAP HANA database

Category	Type	Description
Character	VARCHAR	Stores specified fixed-length character strings.
	NVARCHAR	Variable-length character string contains data in Unicode format.
	TEXT	Stores long character strings. The maximum length of a character string is 2 GB minus 1 byte. Long text strings are stored.
Value	BIGINT	Stores signed integers. Integer part: 19 digits; decimal part: 0 digits
	TINYINT	Stores signed integers. Integer part: 3 digits; decimal part: 0 digits
	SMALLINT	Data of the smallint type occupies two bytes of storage space. One binary bit indicates a positive or negative sign of an integer value, and the other 15 binary bits indicate a length and a size, and may represent all integers ranging from -2^{15} to 2^{15}.
	REAL	The value can be a positive or negative decimal number.
	DECIMAL	Numeric data type with fixed precision and scale
	FLOAT	Stores floating-point numbers with binary precision.
	DOUBLE	Stores double-precision floating-point numbers.
Time and date	DATE	Stores information about the year, month, and day.
	TIME	Stores information about the hour, minute, and second.
	TIMESTAMP	Stores information about the year, month, day, hour, minute, and second.
Multimedia	CLOB	Stores variable-length binary large objects with a maximum length of 2 GB minus 1 byte.
	NCLOB	This type can store a maximum of 4 GB data. When the character set is converted, this type is affected.
Boolean	BOOLEAN	The value is stored in one byte and can be **1**, **0**, or **NULL**.

Data Types Supported in DLI Database Migration

If the migration source is a DLI database, the following data types are supported.

Table 4-13 Data types supported for the DLI database

Category	Type	Description
Character	CHAR	Stores specified fixed-length character strings.
	VARCHAR	Same as CHAR
	STRING	Stores long character strings. The maximum length of a character string is 2 GB minus 1 byte. Long text strings are stored.
Value	BIGINT	Stores signed integers. Integer part: 19 digits; decimal part: 0 digits
	TINYINT	Stores signed integers. Integer part: 3 digits; decimal part: 0 digits
	SMALLINT	Data of the smallint type occupies two bytes of storage space. One binary bit indicates a positive or negative sign of an integer value, and the other 15 binary bits indicate a length and a size, and may represent all integers ranging from -2^{15} to 2^{15}.
	INT	Stores signed integers. Integer part: 10 digits; decimal part: 0 digits
	DECIMAL	Numeric data type with fixed precision and scale
	FLOAT	Stores floating-point numbers with binary precision.
	DOUBLE	Stores double-precision floating-point numbers.
Time and date	DATE	Stores information about the year, month, and day.
	TIMESTAMP	Stores information about the year, month, day, hour, minute, and second.
Boolean	BOOLEAN	The value is stored in one byte and can be **1**, **0**, or **NULL**.

Data Types Supported in Elasticsearch/CSS Database Migration

If the migration source is an Elasticsearch/CSS database, the following data types are supported.

Table 4-14 Data types supported for the Elasticsearch/CSS database

Category	Type	Description	Storage Format Example	MySQL
Character	keyword	Stores strings.	"keyword"	String

Category	Type	Description	Storage Format Example	MySQL
	text	Stores long character strings. The maximum length of a character string is 2 GB minus 1 byte. Long text strings are stored.	"long string"	TEXT
	string	Stores long character strings. The maximum length of a character string is 2 GB minus 1 byte. Long text strings are stored.	"a string"	String
Integer	short	Stores 16-bit signed integers ranging from –32768 to 32767.	32765	smallInt
	integer	Stores 32-bit signed integers ranging from -2^{31} to $2^{31} - 1$.	3276566	int
	long	Stores 64-bit signed integers ranging from -2^{63} to $2^{63} - 1$.	3276566666	BIGINT
Value	double	64-bit IEEE 754 double-precision floating-point format	21.333	double
	float	32-bit IEEE 754 single-precision floating-point format	21.333	double
Boolean	boolean	The value is stored in one byte and can be **1**, **0**, or **NULL**.	1	Boolean
Object	object	A string of flat storage objects	{"users.name": ["John","Smith"], users.age": [26,28], "users.sex" :[1,2]}	TEXT

Category	Type	Description	Storage Format Example	MySQL
Nested	nested	A string of nested storage objects	{"users.name" : "John" , "users.age" : 26, "users.sex" : 1} { "users.name" : "Smith", "users.age" : 28, "users.sex" : 2}	TEXT
Date	date	A string in the date format	"2018-01-13" or "2018-01-13 12:10:30"	DATE or timeStamp
Special type	ip	A string in the IP address format	"192.168.127.100"	String
Array	string_array	An array of strings	["str","str"]	TEXT
	short_array	An array of 16-bit integers	[1,1,1]	TEXT
	integer_array	An array of 32-bit integers	[1,1,1]	TEXT
	long_array	An array of 64-bit integers	[1,1,1]	TEXT
	float_array	An array of 32-bit floating-point numbers	[1.0,1.0,1.0]	TEXT
	double_array	An array of 64-bit floating-point numbers	[1.0,1.0,1.0]	TEXT
Value range	completion	A string that is automatically completed	"string"	TEXT

Data Types Supported in Dameng Database Migration

When the source end is a Dameng database and the destination end is a Hive or DWS database, the following data types are supported.

Table 4-15 Data types supported for the Dameng database

Category	Type	Description	Storage Format Example	Hive	DWS
Character	CHAR	Stores specified fixed-length character strings.	'a' or 'aaaaa'	CHAR	CHAR
	CHARACTER	Same as CHAR	'a' or 'aaaaa'	CHAR	CHAR
	VARCHAR	Stores specified variable-length character strings.	'a' or 'aaaaa'	VARCHAR	VARCHAR
	VARCHAR2	Same as VARCHAR	'a' or 'aaaaa'	VARCHAR	VARCHAR
Value	NUMERIC	Stores zero, positive, and negative fixed point numbers.	52.36	DECIMAL	NUMERIC
	DECIMAL	Similar to NUMERIC	52.36	DECIMAL	NUMERIC
	DEC	Same as DECIMAL	52.36	DECIMAL	NUMERIC
	NUMBER	Same as NUMERIC	52.36	DECIMAL	NUMERIC
	INTEGER	Stores signed integers. Integer part: 10 digits; decimal part: 0 digits	5236	INT	INTEGER
	INT	Same as INTEGER	5236	INT	INTEGER
	BIGINT	Stores signed integers. Integer part: 19 digits; decimal part: 0 digits	5236	BIGINT	BIGINT

Category	Type	Description	Storage Format Example	Hive	DWS
	TINYINT	Stores signed integers. Integer part: 3 digits; decimal part: 0 digits	100	TINYINT	SMALLINT
	SMALLINT	Stores signed integers. Integer part: 5 digits; decimal part: 0 digits	9999	SMALLINT	SMALLINT
	BYTE	Similar to TINYINT. Integer part: 3 digits; decimal part: 0 digits	100	TINYINT	SMALLINT
	BINARY	Stores fixed-length binary data.	0x2A3B4058	BINARY (NULL)	BYTEA (NULL)
	VARBINARY	Stores variable-length binary data.	0x2A3B4058	BINARY (NULL)	BYTEA (NULL)
	FLOAT	Stores floating-point numbers with binary precision.	52.36	FLOAT	FLOAT8
	DOUBLE	Similar to FLOAT	52.36	DOUBLE	FLOAT8
	REAL	Stores binary floating-point numbers.	52.3	FLOAT	FLOAT4
	DOUBLE PRECISION	Stores double-precision floating-point numbers.	52.3	DOUBLE	FLOAT8
Bit string	BIT	Stores 1, 0, or NULL.	1, 0, or NULL	TINYINT(1 0 NULL)	BOOLEAN (true false NULL)
Time and date	DATE	Stores information about the year, month, and day.	'1999-10-01', '1999/10/01', or '1999.10.01'	DATE	TIMESTAMP

Category	Type	Description	Storage Format Example	Hive	DWS
	TIME	Stores information about the hour, minute, and second.	'09:10:21' or '9:10:21'	Not supported (string)	TIME
	TIMESTAMP	Stores information about the year, month, day, hour, minute, and second.	'2002-12-12 09:10:21', '2002-12-12 9:10:21', '2002/12/12 09:10:21', or '2002.12.12 09:10:21'	TIMESTAMP	TIMESTAMP
	TIME WITH TIME ZONE	Stores a TIME value with a time zone. Add the time zone information to the end of the TIME type.	'09:10:21 +8:00', '09:10:21+8:00', or '9:10:21+8:00'	Not supported (string)	TIME WITH TIME ZONE
	TIMESTAMP WITH TIME ZONE	Stores a TIMESTAMP value with a time zone. Add the time zone information to the end of the TIMESTAMP type.	'2002-12-12 09:10:21 +8:00', '2002-12-12 9:10:21 +8:00', '2002/12/12 09:10:21 +8:00', or '2002.12.12 09:10:21 +8:00'	TIMESTAMP	TIMESTAMP WITH TIME ZONE
	TIMESTAMP WITH LOCAL TIME ZONE	Stores the TIMESTAMP value of a local time zone. The standard time zone type (TIMESTAMP WITH TIME ZONE) can be converted to the local time zone type.	'2002-12-12 09:10:21 +8:00', '2002-12-12 9:10:21 +8:00', '2002/12/12 09:10:21 +8:00', or '2002.12.12 09:10:21 +8:00'	Not supported (string)	Not supported (TEXT)

Category	Type	Description	Storage Format Example	Hive	DWS
	DATETIME WITH TIME ZONE	Same as TIMESTAMP WITH TIME ZONE	'2002-12-12 09:10:21 +8:00', '2002-12-12 9:10:21 +8:00', '2002/12/12 09:10:21 +8:00', or '2002.12.12 09:10:21 +8:00'	TIMESTAMP	TIMESTAMP WITH TIME ZONE
	INTERVAL YEAR	Interval of years. The leading precision specifies the range of years.	INTERVAL '0015' YEAR	Not supported (string)	Not supported (VARCHAR)
	INTERVAL YEAR TO MONTH	Interval of months and years. The leading precision specifies the range of years.	INTERVAL '0015-08' YEAR TO MONTH	Not supported (string)	Not supported (VARCHAR)
	INTERVAL MONTH	Interval of months. The leading precision specifies the range of months.	INTERVAL '0015' MONTH	Not supported (string)	Not supported (VARCHAR)
	INTERVAL DAY	Interval of days. The leading precision specifies the range of days.	INTERVAL '150' DAY	Not supported (string)	Not supported (VARCHAR)
	INTERVAL DAY TO HOUR	Interval of hours and days. The leading precision specifies the range of days.	INTERVAL '9 23' DAY TO HOUR	Not supported (string)	Not supported (VARCHAR)
	INTERVAL DAY TO MINUTE	Interval of minutes, hours, and days. The leading precision specifies the range of days.	INTERVAL '09 23:12' DAY TO MINUTE	Not supported (string)	Not supported (VARCHAR)

Category	Type	Description	Storage Format Example	Hive	DWS
	INTERVAL DAY TO SECOND	Interval of seconds, minutes, hours, and days. The leading precision specifies the range of days.	INTERVAL '09 23:12:01.1' DAY TO SECOND	Not supported (string)	Not supported (VARCHAR)
	INTERVAL HOUR	Interval of hours. The leading precision specifies the range of hours.	INTERVAL '150' HOUR	Not supported (string)	Not supported (VARCHAR)
	INTERVAL HOUR TO MINUTE	Interval of minutes and hours. The leading precision specifies the range of hours.	INTERVAL '23:12' HOUR TO MINUTE	Not supported (string)	Not supported (VARCHAR)
	INTERVAL HOUR TO SECOND	Interval of seconds, minutes, and hours. The leading precision specifies the range of hours.	INTERVAL '23:12:01.1' HOUR TO SECOND	Not supported (string)	Not supported (VARCHAR)
	INTERVAL MINUTE	Interval of minutes. The leading precision specifies the range of minutes.	INTERVAL '150' MINUTE	Not supported (string)	Not supported (VARCHAR)
	INTERVAL MINUTE TO SECOND	Interval of minutes and seconds. The leading precision specifies the range of minutes.	INTERVAL '12:01.1' MINUTE TO SECOND	Not supported (string)	Not supported (VARCHAR)
	INTERVAL SECOND	Interval of seconds. The leading precision specifies the value range of the integer part of the second	INTERVAL '51.1' SECOND	Not supported (string)	Not supported (VARCHAR)

Category	Type	Description	Storage Format Example	Hive	DWS
Multimedia	IMAGE	IMAGE specifies the image type in the multimedia information. An image consists of a pixel lattice with a maximum length of 2 GB minus 1 byte. In addition to storing image data, other binary data can also be stored.	0x2A3B4058 (binary data)	Not supported	Not supported
	LONGVARBINARY	Same as IMAGE	0x2A3B4059 (binary data)	Not supported	Not supported
	TEXT	Stores the long string type. The maximum length of a string is 2 GB minus 1 byte.	0x5236 (binary data)	Not supported	Not supported
	LONGVARCHAR	Similar to TEXT	0x5236 (binary data)	Not supported	Not supported
	BLOB	Stores variable-length binary large objects with a maximum length of 2 GB minus 1 byte.	0x5236 (binary data)	Not supported	Not supported
	CLOB	Stores variable-length binary large objects with a maximum length of 2 GB minus 1 byte.	0x5236 (binary data)	Not supported	Not supported

Category	Type	Description	Storage Format Example	Hive	DWS
	BFILE	Specified the binary files stored in the operating systems. Files are stored in the operating systems instead of the databases. They can be read only.	-	Not supported	Not supported

4.4 Managing Clusters

4.4.1 Creating a CDM Cluster

CDM provides independent clusters for secure and reliable data migration. Clusters are isolated from each other and cannot access each other.

CDM clusters can be used in the following scenarios:

- They can be used to create and run data migration jobs.
- They can function as agents for connecting Management Center to a data lake.

Prerequisites

You have applied for a VPC, subnet, and security group. If the CDM cluster tries to connect to another cloud service, ensure that the cluster and the cloud service are in the same VPC. Otherwise, an EIP is required.

NOTE

- If the CDM cluster and a cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other through an intranet.
- If the CDM cluster and the cloud service are in the same region and VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **Configuring Routes**. For details about how to configure security group rules, see **Configuring Security Group Rules**.
- If the CDM cluster and a cloud service are in different VPCs of the same region, you can create a VPC peering connection to enable them to communicate with each other. For details about how to create a VPC peering connection, see **Creating a VPC Peering Connection**.

  Note: If a VPC peering connection is created, the peer VPC subnet may overlap with the CDM management network. As a result, data sources in the peer VPC cannot be accessed. You are advised to use the Internet for cross-VPC data migration, or contact the administrator to add specific routes for the VPC peering connection in the CDM background.
- If the CDM cluster and a cloud service are located in different regions, you need to use the Internet or Direct Connect to enable them to communicate with each other. When using the Internet, ensure that an EIP has been bound to the CDM cluster, the security group of CDM allows outbound traffic from the host where the off-cloud data source is located, the host where the data source is located can access the Internet, and the connection port has been enabled in the firewall rules.
- In addition, an enterprise project may also affect the communication between the CDM cluster and other cloud services. The CDM cluster can communicate with a cloud service only if they have the same enterprise project.

Procedure

If a DataArts Studio instance includes a CDM cluster (except the trial version) and the cluster meets your requirements, you do not need to buy a DataArts Migration incremental package. If you need to create another CDM cluster, buy a DataArts Studio incremental package by referring to **Buying a DataArts Studio Incremental Package**.

4.4.2 Binding or Unbinding an EIP

Scenario

After creating a CDM cluster, you can bind an EIP to or unbind an EIP from the cluster. The EIP is billed based on the VPC service.

If CDM needs to access a local or Internet data source, or a cloud service in another VPC, bind an EIP to the CDM cluster or use a NAT gateway to enable the CDM cluster to share the EIP with ECSs to access the Internet. For details, see **Adding an SNAT Rule**.

NOTE

If SSL encryption is configured for the access channel of a local data source, CDM cannot connect to the data source using the EIP.

Prerequisites

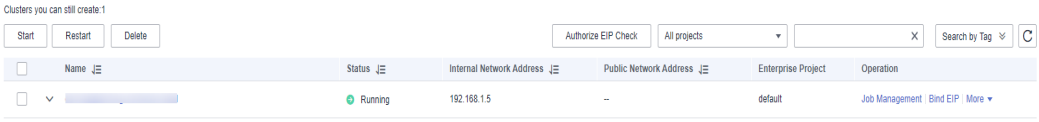
- You have created a CDM cluster.

- Your EIP quota is sufficient.

Procedure

Step 1 Log in to the CDM console and choose **Cluster Management** in the left navigation pane.

Figure 4-3 Cluster list



NOTE

The **Source** column is displayed only when you access the **DataArts Migration** page from the DataArts Studio console.

Step 2 Bind an EIP to or unbind an EIP from a cluster.

- Binding an EIP: In the **Operation** column, click **Bind EIP**. The **Bind EIP** dialog box is displayed.
- Unbinding an EIP: In the **Operation** column, choose **More** > **Unbind EIP**.

Step 3 Click **Yes**.

----End

4.4.3 Restarting a Cluster

Scenario

After modifying some configurations (for example, disabling user isolation), you must restart the cluster to make the modification take effect.

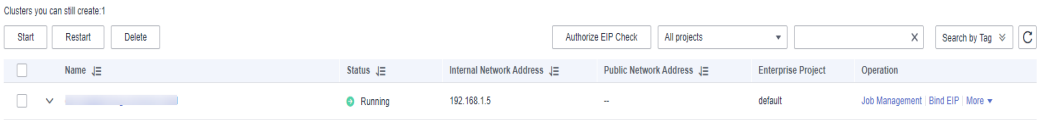
Prerequisites

You have created a CDM cluster.

Restarting a cluster

Step 1 Log in to the CDM console and choose **Cluster Management** in the left navigation pane.

Figure 4-4 Cluster list

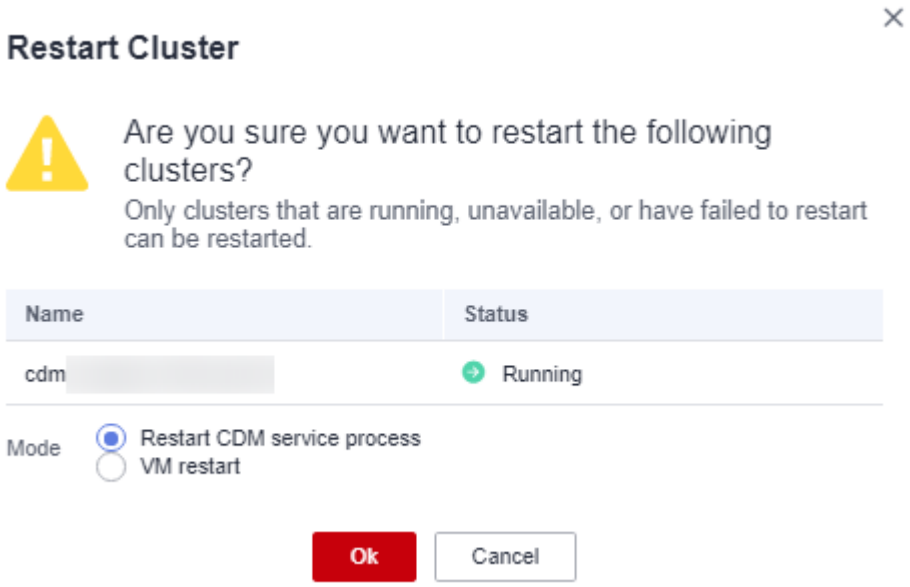


NOTE

The **Source** column is displayed only when you access the **DataArts Migration** page from the DataArts Studio console.

Step 2 Locate the row that contains the target cluster, click **More** in the **Operation** column, and select **Restart** from the drop-down list.

Figure 4-5 Restarting a cluster



Step 3 Select **Restart CDM service process** or **VM restart** and click **OK**.

- **Restart CDM service process**: Only the CDM service process is restarted. The cluster VM will not be restarted.
- **VM restart**: The service process will be interrupted and VMs in the cluster will be restarted.

----End

4.4.4 Deleting a Cluster

Scenario

You can delete a CDM cluster that you no longer use.

CAUTION

After a CDM cluster is deleted, the cluster and its data are destroyed and cannot be restored. Exercise caution when performing this operation.

Before deleting a cluster, note the following:

- Ensure that the cluster is not in use.

- Ensure that the links and jobs in the cluster have been backed up through the job export function described in **Managing Jobs in Batches**.
- You are not advised to delete the CDM cluster which is free of charge. If you delete it, you can only purchase clusters.
- After a CDM cluster is deleted, it will not be billed in pay-per-use mode and the package duration will not be deducted. If you have purchased a CDM discount package or a yearly/monthly CDM incremental package for the CDM cluster to delete, unsubscribe from the package by following the instructions in **Unsubscriptions**.

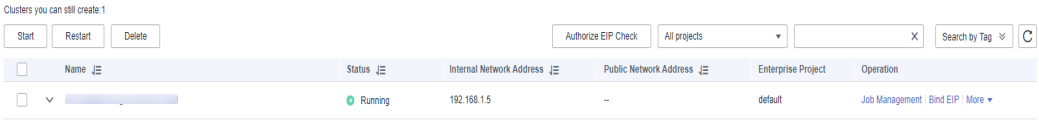
Prerequisites

You have created a CDM cluster.

Deleting a Cluster

Step 1 Log in to the CDM console and choose **Cluster Management** in the left navigation pane.

Figure 4-6 Cluster list



NOTE

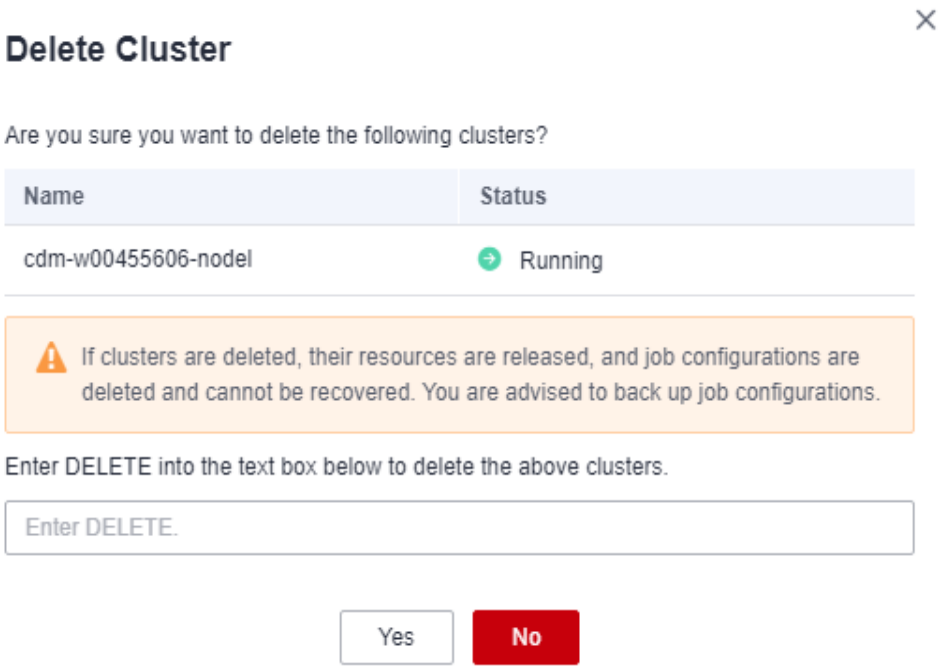
The **Source** column is displayed only when you access the **DataArts Migration** page from the DataArts Studio console.

Step 2 Delete a cluster using either of the following methods:

- Locate a cluster, click **More** in the **Operation** column, and select **Delete**.
- Select a cluster and click **Delete** above the cluster list.

Step 3 Enter **DELETE** and click **Yes**.

Figure 4-7 Deleting a cluster



----End

4.4.5 Downloading Cluster Logs

Scenario

This section describes how to obtain cluster logs to view the job running history and locate job failure causes.

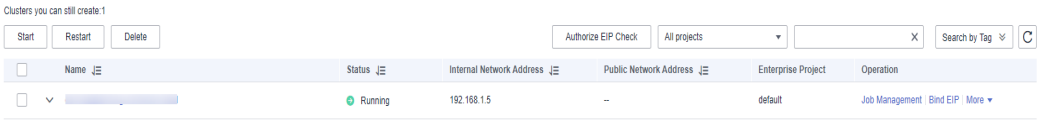
Prerequisites

You have created a CDM cluster.

Procedure

Step 1 Log in to the CDM console and choose **Cluster Management** in the left navigation pane.

Figure 4-8 Cluster list

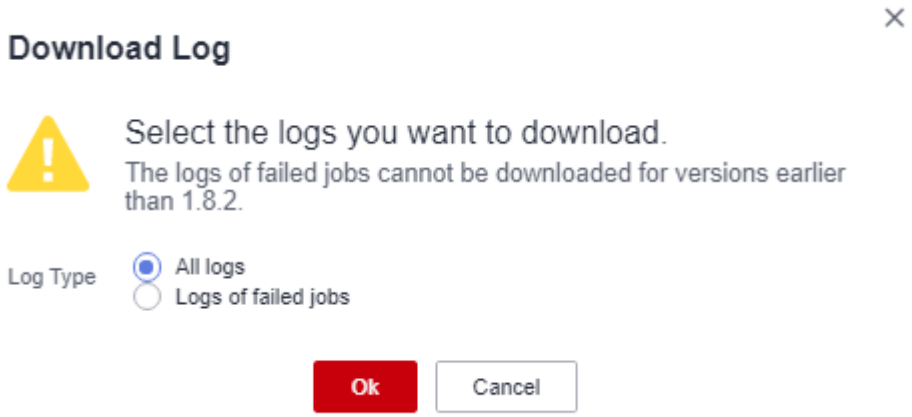


NOTE

The **Source** column is displayed only when you access the **DataArts Migration** page from the DataArts Studio console.

Step 2 Locate the row that contains a cluster, click **More** in the **Operation** column, and select **Download Log** from the drop-down list.

Figure 4-9 Download Log



Step 3 In the displayed dialog box, click **OK** to download logs to a local PC.

----End

4.4.6 Viewing Basic Cluster Information and Modifying Cluster Configurations

Scenario

After creating a CDM cluster, you can view its basic information and modify its configurations.

- You can view the following basic cluster information:
  - **Cluster Information**: cluster version, creation time, project ID, instance ID, and cluster ID
  - **Instance Configuration**: cluster flavor, CPU, and memory
  - **Network**
- You can modify the following cluster configurations:
  - **Notification**: If a CDM migration job (only table/file migration) fails or the EIP is abnormal, CDM sends an SMS or email notification to the user. Notifications generated by this function will not be charged.
  - **User Isolation**: determines whether other users can view and operate the migration jobs and links in the cluster.
    - If this function is enabled, migration jobs and links in the cluster are isolated. Other IAM users of the a Huawei account cannot view or operate the migration jobs and links in the cluster.

NOTE

Starting jobs by group will run all jobs in the group. If user isolation is enabled, starting jobs by group will still run all jobs in the group even if otherIAM users in the a Huawei account cannot view the jobs in the group. Therefore, you are not advised to start jobs by group in user isolation scenarios.

- If this function is disabled, migration jobs and links in the cluster can be shared with other users. All IAM users with the required permission in the a Huawei account can view and operate migration jobs and links.

  After disabling **User Isolation**, restart the cluster VM for the settings to take effect.

- **Maximum Concurrent Extractors**: This parameter specifies the total number of concurrent extractors of a job. If the total number of concurrent extractors of all jobs exceeds the upper limit, the excess extractors will wait in a queue.

  The value of this parameter ranges from 1 to 1000. You are advised to set it based on the cluster specifications. For details about the recommended value, see **Maximum Concurrent Extractors**. If the number of concurrent extractors is too large, memory overflow may occur. Exercise caution when changing the value.

  NOTE

  This parameter is also available on the **Settings** tab page. You can change its value either on this page or the **Settings** page.

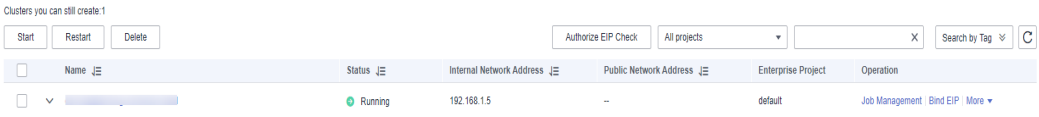
Prerequisites

You have created a CDM cluster.

Viewing Basic Cluster Information

Step 1 Log in to the CDM console and choose **Cluster Management** in the left navigation pane.

Figure 4-10 Cluster list



NOTE

The **Source** column is displayed only when you access the **DataArts Migration** page from the DataArts Studio console.

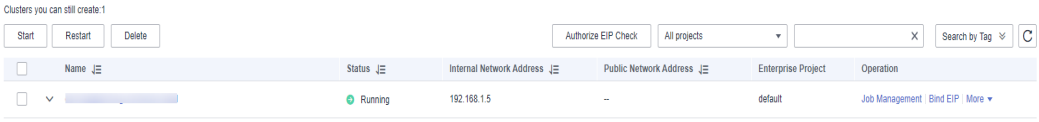
Step 2 Click the cluster name to view its basic information.

----End

Modifying Cluster Configurations

Step 1 Log in to the CDM console and choose **Cluster Management** in the left navigation pane.

Figure 4-11 Cluster list



NOTE

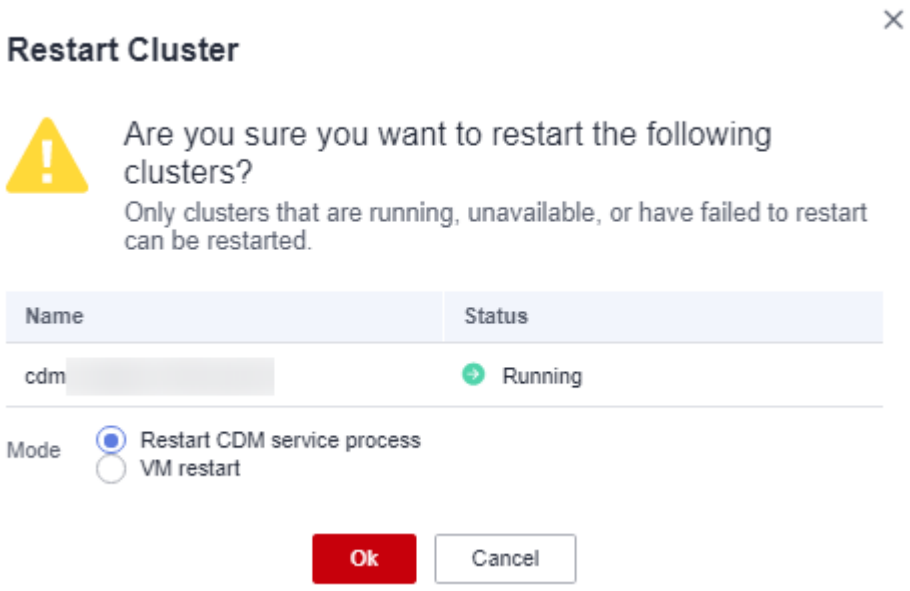
The **Source** column is displayed only when you access the **DataArts Migration** page from the DataArts Studio console.

Step 2 Click the name of a cluster and click the **Cluster Configuration** tab to modify **Notification**, **User Isolation** and **Maximum Concurrent Extractors**.

Step 3 Click **Save**. The **Cluster Management** page is displayed.

Step 4 If **User Isolation** is disabled, choose **More** > **Restart** in the **Operation** column to restart the cluster VM for the settings to take effect.

Figure 4-12 Restarting a cluster



- **Restart CDM service process**: Only the CDM service process is restarted. The cluster VM will not be restarted.
- **VM restart**: The service process will be interrupted and VMs in the cluster will be restarted.

Step 5 Select **VM restart** and click **Yes**.

----End

4.4.7 Managing Cluster Tags

Scenario

You can add, modify, and delete tags for CDM clusters. Tags can be used to identify multiple types of cloud resources. Cloud resources with the same tag can be filtered out in the TMS tag system or on the CDM **Cluster Management** page.

NOTE

A maximum of 10 tags can be added to a CDM cluster.

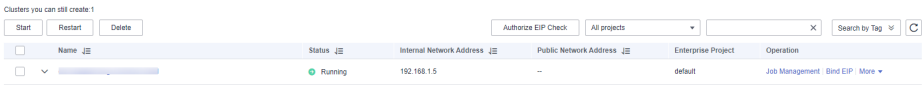
Prerequisites

You have created a CDM cluster.

Procedure

Step 1 Log in to the CDM console. In the navigation pane, choose **Cluster Management**.

Figure 4-13 Cluster list

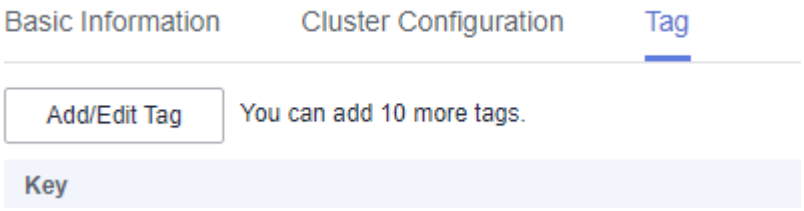


NOTE

The **Source** column is displayed only when you access the **DataArts Migration** page from the DataArts Studio console.

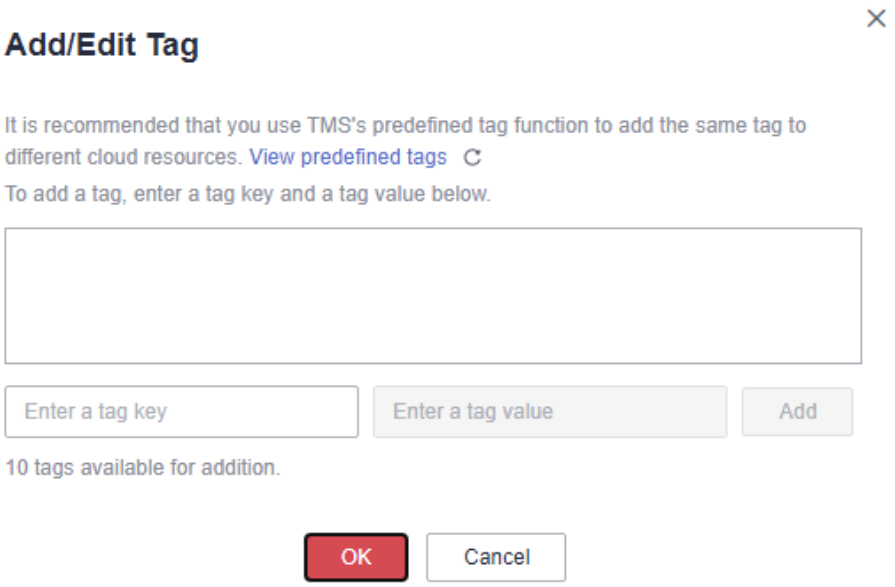
Step 2 Click a cluster name and then the **Tag** tab.

Figure 4-14 Modifying Cluster Configurations



Step 3 Click **Add/Edit Tag** and add tags to or modify tags for the CDM cluster.

Figure 4-15 Adding/Editing a tag



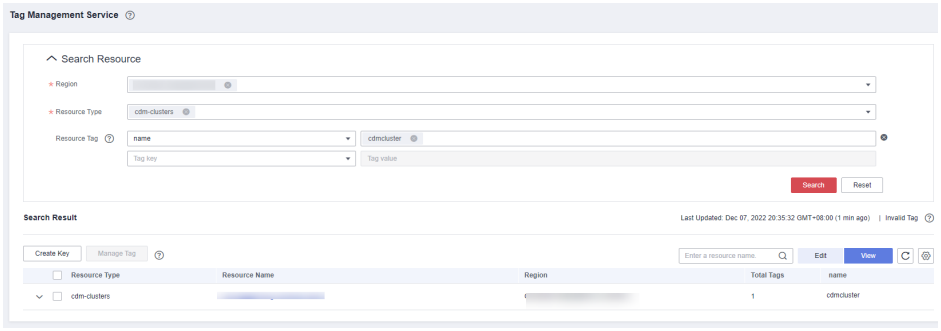
NOTE

- A cluster can have a maximum of 10 tags.
- A tag key and a tag value can contain a maximum of 36 and 43 characters, respectively.

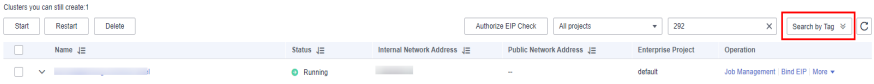
Step 4 (Optional) In the tag list, click **Delete** in the **Operation** column to delete tags.

Step 5 Use either of the following methods to filter out the resources matching specified tags:

- On the TMS console, set resource search criteria and click **Search** to obtain the clusters with the specified tags.



- On the **Cluster Management** page, click **Search by Tag**, select tags, and click **Search** to obtain the clusters with the specified tags.



----End

4.4.8 Viewing Metrics

4.4.8.1 CDM Metrics

Function

Cloud Eye monitors the running status of cloud services and usage of each metric, and creates alarm rules for monitoring metrics.

After you create a CDM cluster, Cloud Eye automatically associates with CDM monitoring metrics to help you understand the running status of the CDM cluster.

- This section describes the CDM metrics that can be monitored by Cloud Eye as well as their namespaces and dimensions.
- For details about CDM monitoring metrics, see **Querying Metrics**.
- For details about how to set alarm rules, see **Configuring Alarm Rules**.

Prerequisites

You have obtained required Cloud Eye permissions.

Namespace

SYS.CDM

Metrics

Table 4-16 lists the CDM metrics.

Table 4-16 CDM metrics

ID	Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
bytes_in	Bytes In	Measures the network inbound rate of the monitored object. Unit: byte/s	≥ 0 bytes/s	Cloud Data Migration	1 minute
bytes_out	Bytes Out	Measures the network outbound rate of the monitored object. Unit: byte/s	≥ 0 bytes/s	Cloud Data Migration	1 minute
cpu_usage	CPU Usage	Measures the CPU usage of the monitored object. Unit: %	0% to 100%	Cloud Data Migration	1 minute

ID	Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
mem_usage	Memory Usage	Measures the memory usage of the monitored object. Unit: %	0% to 100%	Cloud Data Migration	1 minute
pg_pending_job	Number of Queued Jobs	Number of jobs in the PENDING state in the CDM instance. Unit: count	>=0	Cloud Data Migration	1 minute
pending_threads	Maximum Concurrent Extractors	Number of concurrent extraction threads in the Waiting state in the CDM instance. Unit: count	>=0	Cloud Data Migration	1 minute
disk_usage	Disk Usage	Measures the disk usage of the physical server accommodating the monitored ECS, which is not accurate as that obtained on the monitored ECS. Unit: %	0.001% to 90%	Cloud Data Migration	1 minute
disk_io	Disk I/O	Measures the bytes read from and written to a disk per second on the physical server accommodating the monitored ECS, which is not accurate as those obtained on the monitored ECS. Unit: Byte/s	0 GB to 10 GB	Cloud Data Migration	1 minute
tomcat_heap_usage	Heap Memory Usage	Measures the heap memory usage of the physical server accommodating the monitored ECS, which is not accurate as that obtained on the monitored ECS. Unit: %	0.001% to 90%	Cloud Data Migration	1 minute

ID	Name	Description	Value Range	Monitored Object	Monitoring Period (Raw Data)
tomcat_connect	Tomcat Concurrent Connections	Measures the number of Tomcat concurrent connections on the physical server.	0 to 2,147,483,647	Cloud Data Migration	1 minute
tomcat_thread_count	Tomat Threads	Measures the number of Tomcat threads on the physical server.	0 to 2,147,483,647	Cloud Data Migration	1 minute
pg_connect	Database Connections	Measures the number of Postgres database connections on the physical server.	0 to 2,147,483,647	Cloud Data Migration	1 minute
pg_submission_row	Rows	Measures the number of rows in the submission table of the Postgres database on the physical server.	0 to 2,147,483,647	Cloud Data Migration	1 minute
pg_failed_job_rate	Job Failure Rate	Measures the job failure rate of the sqoop process on the physical server. Unit: %	0.001% to 100%	Cloud Data Migration	1 minute
inodes_usage	Inodes Usage	Measures the disk inodes usage of the physical server accommodating the monitored ECS, which is not accurate as that obtained on the monitored ECS. Unit: %	0.001% to 100%	Cloud Data Migration	1 minute

Dimension

Key	Value
instance_id	CDM instance

4.4.8.2 Configuring Alarm Rules

Scenario

Set the alarm rules to customize the monitored objects and notification policies. Then, learn CDM running status in a timely manner.

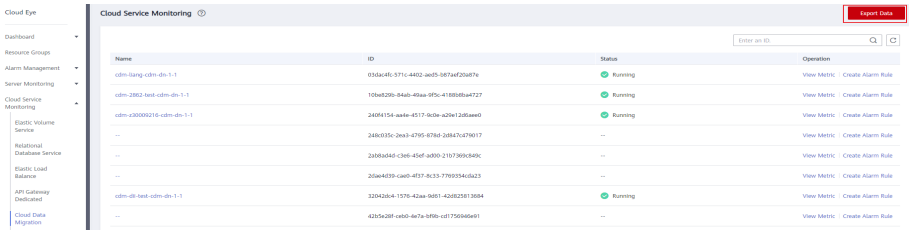
A CDM alarm rule includes the alarm rule name, monitored object, metric, threshold, monitoring interval, and whether to send a notification. This section describes how to set CDM alarm rules.

Procedure

Step 1 Access the CDM console, choose **Cluster Management**. Locate a cluster, click **More** in the **Operation** column, and select **View Metric** from the drop-down list.

Step 2 In the navigation pane, choose **Cloud Service Monitoring** > **Cloud Data Migration**. In the right pane, locate a CDM cluster and click **Create Alarm Rule** in the **Operation** column.

Figure 4-16 Monitored CDM clusters



Step 3 Set the alarm rule for the CDM cluster as prompted.

Step 4 After the setting is complete, click **Confirm**. When an alarm that meets the rule is generated, the system automatically sends a notification.

NOTE

For more information about monitoring and alarms, see the ***Cloud Eye User Guide***.

----End

4.4.8.3 Querying Metrics

Scenario

You can use Cloud Eye to monitor the running status of a CDM cluster. You can view the monitoring metrics on the Cloud Eye console.

Monitored data takes some time for transmission and display. The status displayed on the Cloud Eye console is the status obtained 5 to 10 minutes before. You can view the monitored data of a newly created CDM cluster 5 to 10 minutes later.

Prerequisites

- The CDM cluster is running properly.

If a cluster fails to be restarted or is unavailable, its monitoring metrics are unavailable. You can view the monitored data only after the cluster is restarted or recovered.

- The cluster has been properly running for about 10 minutes.

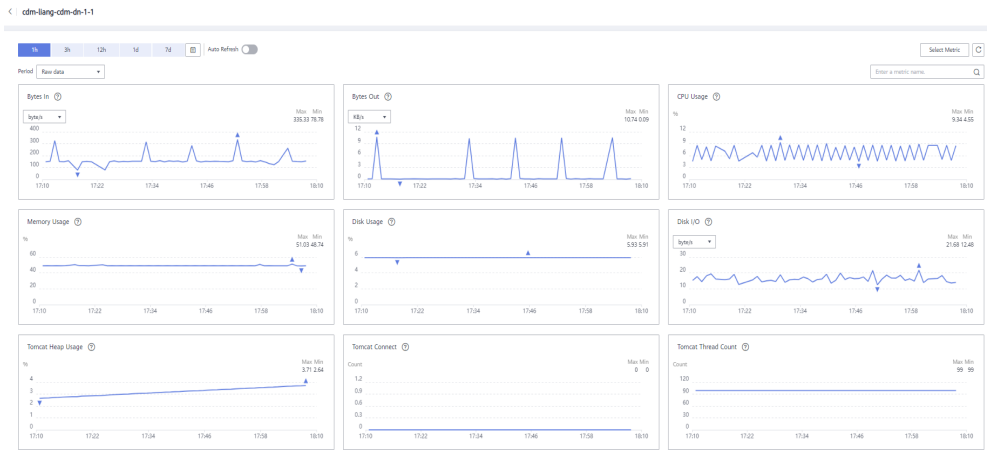
  The monitored data and graphs are available for a newly created cluster after the cluster runs for at least 10 minutes.

Procedure

Step 1 Access the CDM console, choose **Cluster Management**. Locate a cluster, click **More** in the **Operation** column, and select **View Metric** from the drop-down list.

Step 2 On the CDM monitoring page, you can view the graphs of all monitoring metrics.

Figure 4-17 Querying Metrics



Step 3 Click in the upper right corner of the graphs to zoom in the graphs.

Step 4 You can select a time period in the upper left corner to view metric changes in this time period.

----End

4.5 Managing Links

4.5.1 Creating a Link

Scenario

Before creating a data migration job, create a link to enable the CDM cluster to read data from and write data to a data source. A migration job requires a source link and a destination link. For details on the data sources that can be exported (source links) and imported (destination links) in different migration modes (table/file migration), see **Supported Data Sources**.

The link configurations depend on the data source. This section describes how to create these links.

Constraints

- If changes occur in the connected data source (for example, the MRS cluster capacity is expanded), you need to edit and save the connection.
- Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

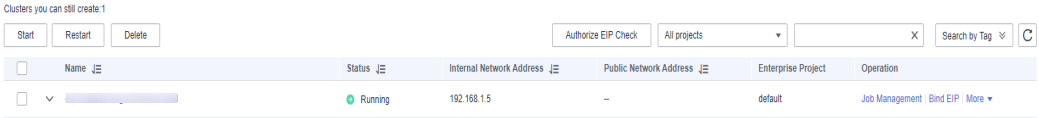
Prerequisites

- A CDM cluster is available.
- The CDM cluster can communicate with the destination data source.
  - If the destination data source is an on-premises database, you need the Internet or Direct Connect. When using the Internet, ensure that an EIP has been bound to the CDM cluster, the security group of CDM allows outbound traffic from the host where the off-cloud data source is located, the host where the data source is located can access the Internet, and the connection port has been enabled in the firewall rules.
  - If the destination data source is a cloud service (such as DWS, MRS, and ECS), the following requirements must be met for network interconnection:
    - If the CDM cluster and the cloud service are in different regions, a public network or a dedicated connection is required for enabling communication between the CDM cluster and the cloud service. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, the host where the data source is located can access the Internet, and the port has been enabled in the firewall rules.
    - If the CDM cluster and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**.
    - The cloud service instance and the CDM cluster belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.
- You have obtained the URL and the account for accessing the data source. The account is granted with the read and write permissions for the data source.

Creating Links

Step 1 Log in to the CDM console and choose **Cluster Management** in the left navigation pane.

Figure 4-18 Cluster list



> **NOTE**
>
> The **Source** column is displayed only when you access the **DataArts Migration** page from the DataArts Studio console.

Step 2 On the CDM console, choose **Cluster Management** in the left navigation pane. Locate the row that contains the target cluster and click **Job Management** in the **Operation** column. On the displayed **Links** page, click **Create Link**. On the displayed page shown in **Figure 4-19**, select a connector.

The connectors are classified based on the type of the data source to be connected. All supported data types are displayed.

Figure 4-19 Selecting a connector type

Step 3 Select a data source and click **Next**. The following describes how to create a MySQL link.

The link parameters of different data sources vary. **Table 4-17** describes the link parameters.

Table 4-17 Link parameters

Connector	Description
• RDS for PostgreSQL • RDS for SQL Server • PostgreSQL • Microsoft SQL Server	Because the JDBC drivers used by these relational databases are the same, the parameters to be configured are also the same and are described in **Link to PostgreSQL/ SQLServer**.
Data Warehouse Service	For details about the parameters, see **Link to DWS**.
SAP HANA	For details about the parameters, see **Link to SAP HANA**.

Connector	Description
Dameng database	For details about the parameters, see **Link to a Dameng Database**.
MySQL	For details about the parameters, see **Link to an RDS for MySQL/MySQL Database**.
Oracle	For details about the parameters, see **Link to an Oracle Database**.
Database Sharding	For details about the parameters, see **Link to a Database Shard**.
Object Storage Service (OBS)	For details about the parameters, see **Link to OBS**.
• MRS HDFS • FusionInsight HDFS • Apache HDFS	If the data source is HDFS of MRS, Apache Hadoop, or FusionInsight HD, see **Link to HDFS**.
• MRS HBase • FusionInsight HBase • Apache HBase	If the data source is HBase of MRS, Apache Hadoop, or FusionInsight HD, see **Link to HBase**.
• MRS Hive • FusionInsight Hive • Apache Hive	If the data source is Hive on MRS, Apache Hadoop, or FusionInsight HD, see **Link to Hive**.
CloudTable Service	If the data source is CloudTable, see **Link to CloudTable**.
• FTP • SFTP	If the data source is an FTP or SFTP server, see **Link to an FTP or SFTP Server**.
HTTP	These connectors are used to read files with an HTTP/HTTPS URL, such as reading public files on the third-party object storage system and web disks. When creating an HTTP link, you only need to configure the link name. The URL is configured during job creation.
MongoDB	If the data source is a local MongoDB, see **Link to MongoDB**.
Document Database Service (DDS)	If the data source is DDS, see **Link to DDS**.
• Redis • Distributed Cache Service	If the data source is Redis or DCS, see **Link to Redis**.
• MRS Kafka • Apache Kafka	If the data source is MRS Kafka or Apache Kafka, see **Link to Kafka**.

Connector	Description
Data Ingestion Service	If the data source is DIS, see **Link to DIS**.
Cloud Search Service (CSS) Elasticsearch	If the data source is CSS or Elasticsearch, see **Link to CSS**.
Data Lake Insight	If the data source is DLI, see **Link to DLI**.
DMS Kafka	If the data source is DMS Kafka, see **Link to DMS Kafka**.
Cassandra	If the data source is Cassandra, see **Link to Cassandra**.
MRS Hudi	For details about the parameters, see **Link to MRS Hudi**.
MRS ClickHouse	For details about the parameters, see **Link to MRS ClickHouse**.
Shentong database	For details about the parameters, see **Link to a ShenTong Database**.

NOTE

Currently, the following data sources are in the OBT phase: FusionInsight HDFS, FusionInsight HBase, FusionInsight Hive, SAP HANA, Document Database Service, CloudTable Service, Cassandra, DMS Kafka, Cloud Search Service, Sharding Database, and ShenTong Database.

Step 4 After configuring the parameters of the link, click **Test** to check whether the link is available. Alternatively, click **Save**, and the system checks automatically.

If the network is poor or the data source is too large, the link test may take 30 to 60 seconds.

----End

Managing Links

CDM allows you to perform the following operations on created links:

- Deleting links: You can delete links that are not used by any job.
- Editing a link: You can modify link parameters but cannot reselect the connector. To modify a link, you need to re-enter the password needed to access the data source.
- Testing connectivity: You can test connectivity of a link that has been saved.
- Viewing the JSON file of a link: You can view parameters of a link in a JSON file.
- Editing the JSON file of a link: Modify parameters of a link in a JSON file.
- Viewing the backend link: You can view the backend link corresponding to a link. For example, you can query details about the backend link if it is enabled.

Before managing a link, ensure that the link is not used by any job to avoid affecting job execution. The procedure for managing connections is as follows:

Step 1 Log in to the management console and choose **Service List** > **Cloud Data Migration**. On the CDM console, choose **Cluster Management** in the left navigation pane. Locate the row that contains the target cluster and click **Job Management** in the **Operation** column. On the displayed page, click the **Links** tab.

Step 2 On the **Links** page, locate the link to be modified.

- Deleting a link: Click **Delete** in the **Operation** column to delete a link. Alternatively, select the links that are not used by any job and click **Delete Link** above the list to delete them.
- Editing the link: Click the link name or click **Edit** in the **Operation** column to access the page for modifying the link. When modifying the link, you need to enter the password for logging in to the data source again.
- Testing connectivity of the link: Click **Test Connectivity** in the **Operation** column.
- Viewing the JSON file of the link: In the **Operation** column, choose **More** > **View Link JSON** to view link parameters in JSON format.
- Editing the JSON file of the link: In the **Operation** column, choose **More** > **Edit Link JSON** to modify link parameters in JSON format.
- Viewing the backend link: Locate the row that contains a link and click **More** in the **Operation** column and select **View Backend Link** to view the backend link corresponding to the link.

----End

4.5.2 Managing Drivers

The Java Database Connectivity (JDBC) provides programmatic access to relational databases. Applications can execute SQL statements and retrieve data using the JDBC API.

Before connecting CDM to a relational database, you need to upload the JDK 8 .jar driver of the relational database.

Prerequisites

- A cluster has been created.
- You have downloaded one of the drivers listed in **Table 4-18**.
- (Optional) An SFTP link has been created by referring to **Link to an FTP or SFTP Server** and the corresponding driver has been uploaded to the offline file server.

How Do I Obtain a Driver?

Select a driver version that adapts to the database type. Note that the version of the uploaded driver does not need to match the version of the database to be connected. Obtain the JDK8 .jar driver of the recommended version by referring to **Table 4-18**.

Table 4-18 Drivers

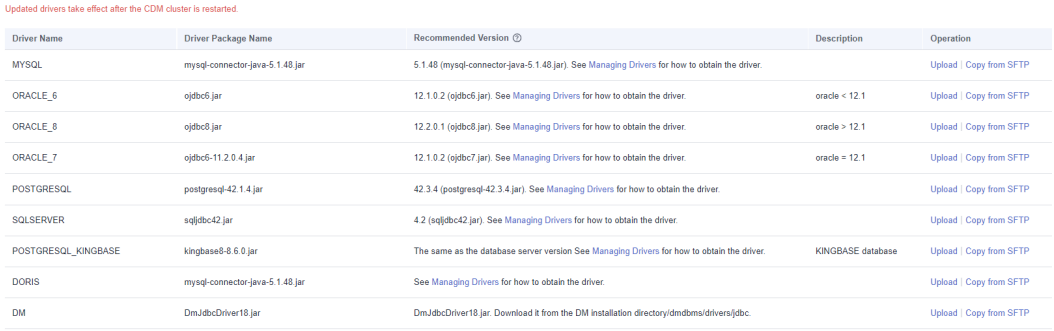
Relational Database Type	Driver Name	How to Obtain	Recommended Version
• RDS for MySQL • MySQL	MySQL	**https://downloads.mysql.com/archives/c-j/**	mysql-connector-java-5.1.48.jar
Oracle	ORACLE_6 ORACLE_7 ORACLE_8	Driver packages: **https://www.oracle.com/database/technologies/appdev/jdbc-downloads.html** Driver packages of historical versions: **https://repo1.maven.org/maven2/com/oracle/database/jdbc/**	ojdbc8.jar for version 12.2.0.1 **NOTE** New versions (for example, Oracle Database 21c (21.3) drivers) are not supported. If they are used, the schema name cannot be obtained during job creation.
• RDS for PostgreSQL • PostgreSQL	POSTGRESQL	**https://mvnrepository.com/artifact/org.postgresql/postgresql**	postgresql-42.3.4.jar for version 42.3.4
KingBase	POSTGRESQL	**https://mvnrepository.com/artifact/org.postgresql/postgresql**	postgresql-42.2.9.jar for PostgreSQL 42.2.9
GaussDB	POSTGRESQL	GaussDB JDBC driver: Search for "JDBC Package, Driver Class, and Environment Class" in **GaussDB Documentation**, select the document corresponding to the instance version, and obtain **gsjdbc4.jar** by referring to the document.	Obtain **gsjdbc4.jar** from the release package of the corresponding version.

Relational Database Type	Driver Name	How to Obtain	Recommended Version
• RDS for SQL Server • Microsoft SQL Server	SQLServer	https://docs.microsoft.com/en-us/sql/connect/jdbc/release-notes-for-the-jdbc-driver?view=sql-server-ver15#previous-releases	sqljdbc42.jar

Procedure

Step 1 Access the CDM console, choose **Cluster Management** in the navigation pane, locate the target cluster, and choose **Job Management** > **Link Management** > **Driver Management**. On the **Driver Management** page, upload a driver.

Figure 4-20 Uploading a driver



Step 2 Click **Upload** in the **Operation** column and select a local driver.

Alternatively, click **Copy from SFTP** in the **Operation** column and configure the **SFTP Link** name and **Driver File Path**.

Step 3 (Optional) If you have uploaded an updated version of a driver, you must restart the CDM cluster for the new driver to take effect.

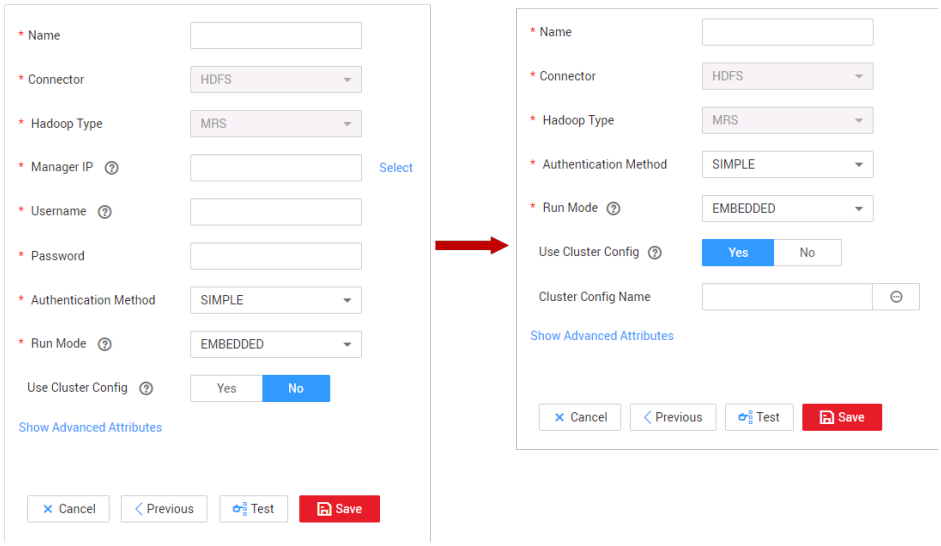
----End

4.5.3 Managing Cluster Configurations

On the **Cluster Configurations** page, you can create, edit, or delete Hadoop cluster configurations.

When creating a Hadoop link, the Hadoop cluster configurations can simplify the link creation. See **Figure 4-21** for details.

Figure 4-21 Comparison before and after using the cluster configurations



CDM supports the following types of Hadoop links:

- MRS clusters: MRS HDFS, MRS HBase, and MRS Hive
- FusionInsight clusters: FusionInsight HDFS, FusionInsight HBase, and FusionInsight Hive
- Apache clusters: Apache HDFS, Apache HBase, and Apache Hive

Scenario

Before creating a Hadoop link, you are advised to create cluster configurations to simplify the link parameter configurations.

Prerequisites

- A cluster has been created.
- You have obtained the Hadoop cluster configuration file and keytab file. See **Table 1** for details.

Obtaining the Cluster Configuration File and Keytab File

The methods for obtaining the Hadoop cluster configuration file and keytab file vary depending on the Hadoop cluster type. For details, see **Table 1**.

Table 4-19 Obtaining the cluster configuration file and keytab file

Hadoop Link	Obtaining the Cluster Configuration File	Obtaining the Keytab File
MRS cluster • MRS HDFS • MRS HBase • MRS Hive • MRS Hudi • MRS ClickHouse	For clusters of MRS 3.x: 1. Log in to FusionInsight Manager. 2. Choose **Cluster** > *Name of the desired cluster* > **Dashboard** > **More** > **Download Client**. 3. In the dialog box that is displayed, select **Configuration Files Only**. The platform type must be the same as that on the server. Retain the default values of other parameters and click **OK** to download the configuration file to the local host. 4. Obtain the downloaded TAR package, which is the FusionInsight cluster configuration file. For clusters of MRS 2.x or earlier: 1. Log in to the MRS console. 2. Choose **Clusters** > **Active Clusters** and click a cluster name to go to the cluster details page. Click the **Components** tab. 3. Click **Download Client**. Set **Client Type** to **Only configuration files**, set **Download To** to **Server** or **Remote host**, customize the client path, and click **OK** to generate the client configuration file. 4. Save the generated configuration file to a local path. See MRS documentation for details.	For clusters of MRS 3.x: 1. Log in to FusionInsight Manager. 2. Choose **System** > **Permission** > **User**, locate the row that contains the target user, and choose **More** > **Download Authentication Credential** to download the authentication credential file. 3. Obtain the downloaded TAR package, which is the keytab file of the FusionInsight cluster. For clusters of MRS 2.x or earlier: 1. Log in to MRS Manager and click **System**. In the **Permission** area, click **Manage User**. 2. In the row of the user for whom you want to export the keytab file, choose **More** > **Download authentication credential** to download the authentication file. After the file is automatically generated, save it to a specified path and keep it properly. See MRS documentation for details.

Hadoop Link	Obtaining the Cluster Configuration File	Obtaining the Keytab File
FusionInsight clusters: • FusionInsight HDFS • FusionInsight HBase • FusionInsight Hive	1. Log in to FusionInsight Manager. 2. Choose **Cluster** > *Name of the desired cluster* > **Dashboard** > **More** > **Download Client**. 3. In the dialog box that is displayed, select **Configuration Files Only**. The platform type must be the same as that on the server. Retain the default values of other parameters and click **OK** to download the configuration file to the local host. 4. Obtain the downloaded TAR package, which is the FusionInsight cluster configuration file. See the FusionInsight documentation for details.	1. Log in to FusionInsight Manager. 2. Choose **System** > **Permission** > **User**, locate the row that contains the target user, and choose **More** > **Download Authentication Credential** to download the authentication credential file. 3. Obtain the downloaded TAR package, which is the keytab file of the FusionInsight cluster. See the FusionInsight documentation for details.

Hadoop Link	Obtaining the Cluster Configuration File	Obtaining the Keytab File
Apache clusters: • Apache HDFS • Apache HBase • Apache Hive	In the Apache cluster scenario, only the required configuration files and packaging rules are described. For details about how to obtain each configuration file, see the corresponding documentation. • HDFS needs to compress the following files into a .zip package without the directory format: – hosts – core-site.xml – hdfs-site.xml – yarn-site.xml – mapred-site.xml – krb5.conf (optional, for clusters in security mode) • HBase needs to compress the following files into a .zip package without the directory format: – hosts – core-site.xml – hdfs-site.xml – yarn-site.xml – mapred-site.xml – hbase-site.xml – krb5.conf (optional, for clusters in security mode) • Hive needs to compress the following files into a .zip package without the directory format: – hosts – core-site.xml – hdfs-site.xml – yarn-site.xml	In the Apache cluster scenario, only the principles for packaging authentication credential files are required. For details about how to obtain the authentication credential files, see the corresponding documentation. 1. Rename the user's authentication credential file as **user.keytab**. 2. Compress the **user.keytab** file into a .zip package without the directory format: **user.keytab.zip**.

Hadoop Link	Obtaining the Cluster Configuration File	Obtaining the Keytab File
	– mapred-site.xml – hive-site.xml – hivemetastore-site.xml – krb5.conf (optional, for clusters in security mode)	

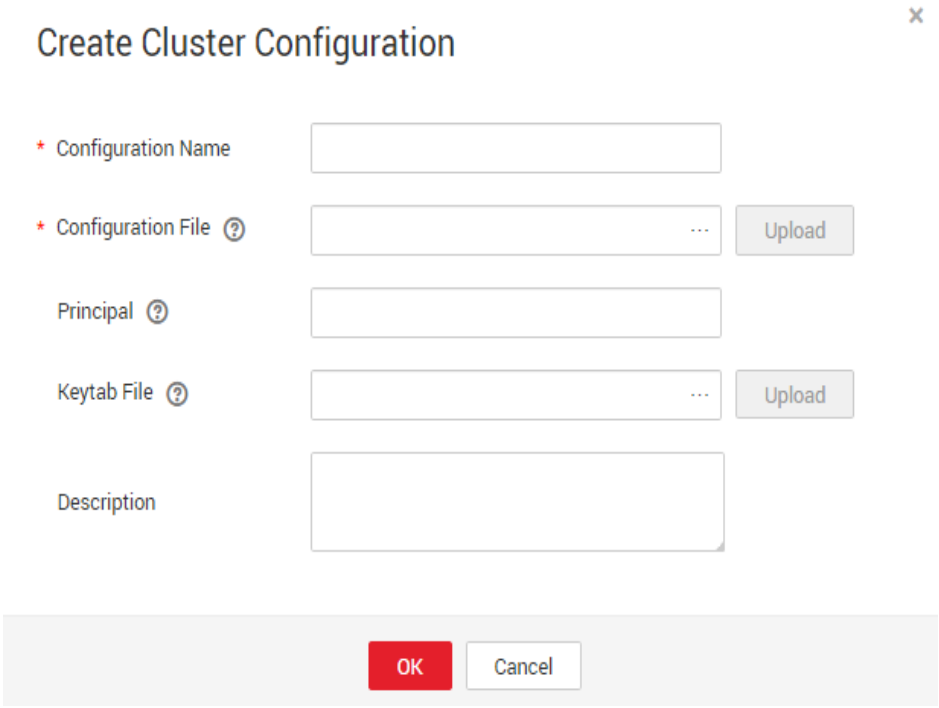
NOTE

- A cluster configuration file contains the configuration parameters of the cluster. If the cluster configuration parameters are modified, you need to obtain the configuration file again.
- The keytab file is the authentication credential file. Before obtaining the keytab file, you need to change the password of this user at least once in the cluster. Otherwise, the downloaded keytab file may be unavailable. After a user password is changed, the exported keytab file becomes invalid, and you need to export a keytab file again.
- The keytab file is used only in a cluster in security mode. In other cases, you do not need to prepare the keytab file.

Procedure

1. On the CDM console, choose **Cluster Management** in the left navigation pane. Locate the row that contains a cluster and choose **Job Management** > **Links** > **Cluster Configurations**.
2. On the **Cluster Configurations** page, click **Create Cluster Configuration** and set the parameters as prompt.

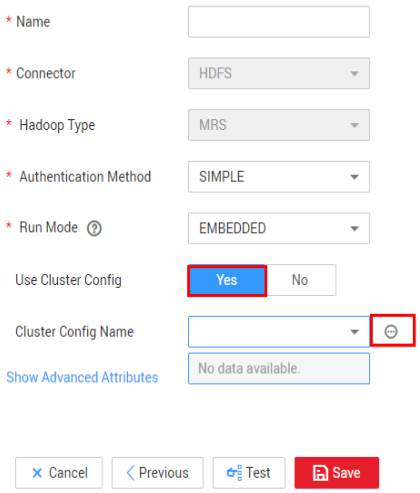
Figure 4-22 Creating cluster configurations



- **Configuration Name**: Enter a cluster configuration name that is easy to remember and distinguish based on the type of the data source to be connected.
- **Configuration File**: Click **Select File** to select a local cluster configuration file, and then click **Upload** on the right to upload the file.
- **Principal**: This parameter is required only for clusters in security mode. Principal is the username in Kerberos security mode and must be the same as that in the keytab file.
- **Keytab File**: Upload the keytab file only for clusters in security mode. Click **Select File** to select a local keytab file, and then click **Upload** on the right to upload the file.
- **Description**: Add a description to identify and distinguish the cluster configuration.

3. Click **OK**. When creating a Hadoop link, set **Authentication Method** as required, **Use Cluster Config** to **Yes**, and then select the corresponding cluster configuration name to quickly create a Hadoop link.

Figure 4-23 Use Cluster Config



4.5.4 Link to OBS

When connecting CDM to the destination OBS bucket, you need to add the read and write permissions to the destination OBS bucket, and file authentication is not required.

NOTE

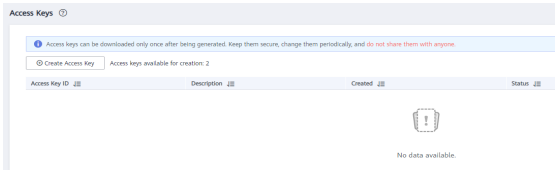
- If the CDM cluster and OBS bucket are not in the same region, the CDM cluster cannot access the OBS bucket.
- Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

When connecting CDM to OBS, configure the parameters as described in **Table 4-20**.

Table 4-20 Parameter description

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	obs_link

Parameter	Description	Example Value
OBS Endpoint	An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions. You can obtain the OBS bucket endpoint by either of the following means: To obtain the endpoint of an OBS bucket, go to the OBS console and click the bucket name to go to its details page. **NOTE** • If the CDM cluster and OBS bucket are not in the same region, the CDM cluster cannot access the OBS bucket. • Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.	-
Port	Data transmission port. The HTTPS port number is 443 and the HTTP port number is 80.	443
OBS Bucket Type	Select a value from the drop-down list, generally, **Object Storage**.	Object Storage

Parameter	Description	Example Value
AK	AK and SK are used to log in to the OBS server. You need to create an access key for the current account and obtain an AK/SK pair. To obtain an access key, perform the following steps: 1. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list. 2. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 4-24**. **Figure 4-24** Clicking Create Access Key  3. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**. **NOTE** • Only two access keys can be added for each user. • To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.	-
SK		-

4.5.5 Link to PostgreSQL/SQLServer

Table 4-21 lists the parameters for creating a link to PostgreSQL/SQLServer. KingBase and GaussDB can be connected through the PostgreSQL connector. The source and destination data sources supported by migration jobs are the same as those for PostgreSQL..

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-21 PostgreSQL/SQLServer link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	sql_link
Database Server	IP address or domain name of the database to connect Click **Select** next to the text box to obtain the list of instances.	192.168.0.1
Port	Port of the database to connect	The port number varies depending on the database. Examples: Default port of SQL Server: **1433** Default port of PostgreSQL: **5432**
Database Name	Name of the database to connect	dbname
Username	Username used for accessing the database This account must have the permissions required to read and write data tables and metadata.	cdm
Password	Password of the user	-
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-
Agent	The agent function will be unavailable soon and does not need to be configured.	-
Driver Version	Different types of relational databases adapt to different drivers. For details, see **How Do I Obtain a Driver?**	-
Fetch Size	(Optional) Displayed when you click **Show Advanced Attributes**. Number of rows obtained by each request. Set this parameter based on the data source and the job's data size. If the value is either too large or too small, the job may run for a long time.	1000

Parameter	Description	Example Value
Link Attributes	(Optional) Click **Add** to add the JDBC connector attributes of multiple specified data sources. For details, see the JDBC connector document of the corresponding database. The following are some examples: • **connectTimeout=60** and **socketTimeout=300**: When a large amount of data needs to be migrated or the entire table is retrieved using query statements, the migration fails due to connection timeout. In this case, you can customize the connection timeout interval (s) and socket timeout interval (s) to prevent failures caused by timeout. • **useCursorFetch=false**: By default, **useCursorFetch** is enabled, indicating that the JDBC connector communicates with relational databases using a binary protocol. Some third-party systems may have compatibility issues, causing migration time conversion errors. In this case, you can disable this function. • **trustServerCertificate=true**: A PKIX error may be reported during the creation of a secure connection. You are advised to set this parameter to **true**.	sslmode=require
Reference Sign	(Optional) Delimiter between the names of the referenced tables or columns. For details, see the product documentation of the corresponding database.	"

4.5.6 Link to DWS

Table 4-22 describes the DWS link parameters.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-22 DWS link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	dws_link

Parameter	Description	Example Value
Database Server	IP address or domain name of the database to connect Click **Select** next to the text box to obtain the list of instances.	192.168.0.1
Port	Port of the database to connect	The port number varies depending on the database.
Database Name	Name of the database to connect	dbname
Username	Username used for accessing the database. This user must have the permissions to read and write data tables and metadata.	cdm
Password	Password of the user	-
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-
Agent	The agent function will be unavailable soon and does not need to be configured.	-
Reference Sign	(Optional) Delimiter between the names of the referenced tables or columns. For details, see the product documentation of the corresponding database.	"
Fetch Size	(Optional) Displayed when you click **Show Advanced Attributes**. Number of rows obtained by each request. Set this parameter based on the data source and the job's data size. If the value is either too large or too small, the job may run for a long time.	1000
SSL Encryption	(Optional) If you set this parameter to **Yes**, CDM can connect to the database (on-premises databases excluded) in SSL encryption mode.	Yes **NOTE** To enable SSL encryption, you must ensure that it is enabled for GaussDB(DWS).

Parameter	Description	Example Value
Link Attributes	(Optional) Click **Add** to add the JDBC connector attributes of multiple specified data sources. For details, see the JDBC connector document of the corresponding database. The following are some examples: • **connectTimeout=60** and **socketTimeout=300**: When a large amount of data needs to be migrated or the entire table is retrieved using query statements, the migration fails due to connection timeout. In this case, you can customize the connection timeout interval (s) and socket timeout interval (s) to prevent failures caused by timeout. • **useCursorFetch=false**: By default, **useCursorFetch** is enabled, indicating that the JDBC connector communicates with relational databases using a binary protocol. Some third-party systems may have compatibility issues, causing migration time conversion errors. In this case, you can disable this function. Open-source MySQL databases support the **useCursorFetch** parameter, and you do not need to set this parameter.	sslmode=require **NOTE** If SSL encryption is enabled but **sslmode** is not set, the link may fail.

4.5.7 Link to an RDS for MySQL/MySQL Database

Table 4-23 lists the parameters for a link to a MySQL database.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-23 MySQL database link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	mysql_link
Database Server	IP address or domain name of the database to connect Click **Select** next to the text box and select a MySQL DB instance in the displayed dialog box.	192.168.0.1

Parameter	Description	Example Value
Port	Port of the database to connect	3306
Database Name	Name of the database to connect	dbname
Username	Username used for accessing the database This account must have the permissions required to read and write data tables and metadata.	cdm
Password	Password of the user	-
Use Local API	(Optional) Whether to use the local API of the database for acceleration. When you create a MySQL link, CDM automatically enables the **local_infile** system variable of the MySQL database to enable the LOAD DATA function, which accelerates data import to the MySQL database. If this parameter is enabled, the date type that does not meet the format requirements will be stored as 0000-00-00. For details, visit the official MySQL website. If CDM fails to enable this function, contact the database administrator to enable the **local_infile** system variable. Alternatively, set **Use Local API** to **No** to disable API acceleration. If data is imported to RDS for MySQL, the LOAD DATA function is disabled by default. In such a case, you need to modify the parameter group of the MySQL instance and set **local_infile** to **ON** to enable the LOAD DATA function. **NOTE** If **local_infile** on RDS is uneditable, it is the default parameter group. You need to create a parameter group, modify its values, and apply it to the RDS for MySQL instance. For details, see the *Relational Database Service User Guide*.	Yes
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-
Agent	The agent function will be unavailable soon and does not need to be configured.	-
local_infile Character Set	When using local_infile to import data to MySQL, you can configure the encoding format.	utf8
Driver Version	Select a driver version that adapts to the database type.	-

Parameter	Description	Example Value
Fetch Size	(Optional) Displayed when you click **Show Advanced Attributes**. Number of rows obtained by each request. Set this parameter based on the data source and the job's data size. If the value is either too large or too small, the job may run for a long time.	1000
Commit Size	(Optional) Displayed when you click **Show Advanced Attributes**. Number of records submitted each time. Set this parameter based on the data destination and the job's data size. If the value is either too large or too small, the job may run for a long time.	-
SSL Encryption	(Optional) If you set this parameter to **Yes**, CDM can connect to the database (on-premises databases excluded) in SSL encryption mode.	Yes

Parameter	Description	Example Value
Link Attributes	(Optional) Click **Add** to add the JDBC connector attributes of multiple specified data sources. For details, see the JDBC connector document of the corresponding database. The following are some examples: ● **connectTimeout=600000** and **socketTimeout=300000**: When a large amount of data needs to be migrated or the entire table is retrieved using query statements, the migration fails due to connection timeout. In this case, you can customize the connection timeout interval (ms) and socket timeout interval (ms) to prevent failures caused by timeout. ● **tinyInt1isBit=false** or **mysql.bool.type.transform=false**: By default, **tinyInt1isBit** is **true**, indicating that **TINYINT(1)** is processed as a bit, that is, **Types.BOOLEAN**, and **1** or **0** is read as **true** or **false**. As a result, the migration fails. In this case, you can set **tinyInt1isBit** to **false** to avoid migration failures. ● **useCursorFetch=false**: By default, **useCursorFetch** is enabled, indicating that the JDBC connector communicates with relational databases using a binary protocol. Some third-party systems may have compatibility issues, causing migration time conversion errors. In this case, you can disable this function. Open-source MySQL databases support the **useCursorFetch** parameter, and you do not need to set this parameter. ● **allowPublicKeyRetrieval=true**: By default, public key retrieval is disabled for MySQL databases. If TLS is unavailable and an RSA public key is used for encryption, connection to an MySQL database may fail. In this case, you can enable public key retrieval to avoid connection failures.	sslmode=require
Reference Sign	(Optional) Delimiter between the names of the referenced tables or columns. For details, see the product documentation of the corresponding database.	`

Parameter	Description	Example Value
Batch Size	Number of rows written each time. It should be less than Commit Size. When the number of rows written reaches the value of Commit Size, the rows will be committed to the database.	100

4.5.8 Link to an Oracle Database

Table 4-24 lists the parameters for a link to an Oracle database.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-24 Oracle database link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	oracle_link
Database Server	IP address or domain name of the database to connect	192.168.0.1
Port	Port of the database to connect	Default port: 1521
Connection Type	Oracle database connection type. The following options are available: • **Service Name**: Use **SERVICE_NAME** to connect to the Oracle database. • **SID**: Use **SID** to connect to the Oracle database.	SID
Instance Name	Oracle instance ID, which is used to differentiate databases by instances. This parameter is available only when **Connection Type** is set to **SID**.	dbname
Database Name	Name of the database to connect This parameter is available only when **Connection Type** is set to **Service Name**.	dbname
Username	Username used for accessing the database This account must have the permissions required to read and write data tables and metadata.	cdm
Password	Password of the username	-
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-

Parameter	Description	Example Value
Agent	The agent function will be unavailable soon and does not need to be configured.	-
Oracle Version	Oracle database version. This parameter is available only for Oracle links. If **java.sql.SQLException: Protocol violation** is displayed, select another version.	Later than 12.1
Fetch Size	(Optional) Displayed when you click **Show Advanced Attributes**. Number of rows obtained by each request. Set this parameter based on the data source and the job's data size. If the value is either too large or too small, the job may run for a long time. A migration from the Oracle to DWS database may time out due to a long data write duration in the DWS database. In this case, reduce the value of **Fetch Size** for the Oracle database.	1000
Link Attributes	(Optional) Click **Add** to add the JDBC connector attributes of multiple specified data sources. For details, see the JDBC connector document of the corresponding database. The following are some examples: • **oracle.net.CONNECT_TIMEOUT=60000** and **oracle.jdbc.ReadTimeout=300000**: When a large amount of data needs to be migrated or the entire table is retrieved using query statements, the migration fails due to connection timeout. In this case, you can customize the connection timeout interval (ms) and read timeout interval (ms) to prevent failures caused by timeout.	sslmode=require
Reference Sign	(Optional) Delimiter between the names of the referenced tables or columns. For details, see the product documentation of the corresponding database.	"
Driver Version	Different types of relational databases adapt to different drivers. For details, see **How Do I Obtain a Driver?**	-

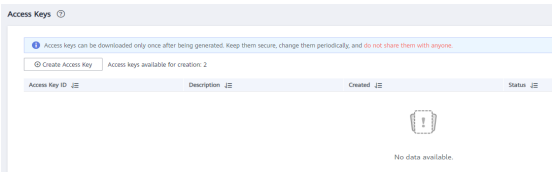
4.5.9 Link to DLI

When connecting CDM to DLI, configure the parameters as described in **Table 4-25**.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-25 DLI link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	dli_link
AK	AK/SK required for authentication during access to the DLI database. You need to create an access key for the current account and obtain an AK/SK pair. 1. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list. 2. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 4-25**. **Figure 4-25** Clicking Create Access Key  3. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**. **NOTE** • Only two access keys can be added for each user. • To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.	-
SK		-

Parameter	Description	Example Value
Project ID	Project ID in the region where DLI resides A project is a group of tenant resources, and an account ID corresponds to the current account. The IAM ID corresponds to the current user. You can view the project IDs, account IDs, and user IDs in different regions on the corresponding pages. 1. Register with and log in to the management console. 2. Hover the cursor on the username in the upper right corner and select **My Credentials** from the drop-down list. 3. On the **API Credentials** page, obtain the account name, account ID, IAM username, and IAM user ID, and obtain the project ID from the project list.	-

4.5.10 Link to Hive

CDM supports the following Hive data sources:

- **MRS Hive**
- **FusionInsight Hive**
- **Apache Hive**

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

MRS Hive

You can view a table during field mapping only when you have the permission to access the table connected to MRS Hive.

MRS Hive links apply to the MapReduce Service (MRS) on Huawei Cloud. **Table 4-26** describes related parameters.

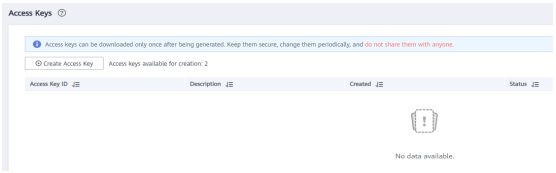
NOTE

- Before creating an MRS Hive link, you need to add an authenticated Kerberos user on MRS and log in to the MRS management page to change the initial password. Then use the new user to create an MRS link.
- To connect to an MRS 2.*x* cluster, create a CDM cluster of version 2.*x* first. CDM 1.8.*x* clusters cannot connect to MRS 2.*x* clusters.
- Currently, the Hive link obtains the **core-site.xml** configuration information from MRS HDFS. Therefore, if MRS Hive uses OBS as the underlying storage system, configure the AK/SK of OBS on MRS HDFS before creating the Hive link.
- Ensure that the MRS cluster and the DataArts Studio instance can communicate with each other. The following requirements must be met for network interconnection:
  - If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule.
  - If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**.
  - The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.

Table 4-26 MRS Hive link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	hivelink
Manager IP	Floating IP address of MRS Manager. Click **Select** next to the **Manager IP** text box to select an MRS cluster. CDM automatically fills in the authentication information.	127.0.0.1
Authentication Method	Authentication method used for accessing MRS • **SIMPLE**: Select this for non-security mode. • **KERBEROS**: Select this for security mode.	SIMPLE
HIVE Version	Set this to the Hive version on the server.	HIVE_3_X

Parameter	Description	Example Value
Username	If **Authentication Method** is set to **KERBEROS**, you must provide the username and password used for logging in to MRS Manager. If you need to create a snapshot when exporting a directory from HDFS, the user configured here must have the administrator permission on HDFS. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user and set **Username** and **Password** to the username and password of the created MRS user when creating an MRS data connection. **NOTE** • If the CDM cluster version is 2.9.0 or later and the MRS cluster version is 3.1.0 or later, the created user must have the permissions of the **Manager_viewer** role to create links on CDM. To perform operations on databases, tables, and columns of an MRS component, you also need to add the database, table, and column permissions of the MRS component to the user by following the instructions in the MRS documentation. • If the CDM cluster version is earlier than 2.9.0 or the MRS cluster version is earlier than 3.1.0, the created user must have the permissions of **Manager_administrator** or **System_administrator** to create links on CDM. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.	cdm
Password	Password used for logging in to MRS Manager	-
Enable ldap	This parameter is available when **Proxy connection** is selected for **Connection Type**. If LDAP authentication is enabled for an external LDAP server connected to MRS Hive, the LDAP username and password are required for authenticating the connection to MRS Hive. In this case, this option must be enabled. Otherwise, the connection will fail.	No
ldapUsername	This parameter is mandatory when **Enable ldap** is enabled. Enter the username configured when LDAP authentication was enabled for MRS Hive.	-

Parameter	Description	Example Value
ldapPassword	This parameter is mandatory when **Enable ldap** is enabled. Enter the password configured when LDAP authentication was enabled for MRS Hive.	-
OBS storage support	The server must support OBS storage. When creating a Hive table, you can store the table in OBS.	No
AK	This parameter is mandatory when **OBS storage support** is enabled. The account corresponding to the AK/SK pair must have the OBS Buckets Viewer permission. Otherwise, OBS cannot be accessed and the "403 AccessDenied" error is reported. You need to create an access key for the current account and obtain an AK/SK pair. 1. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list. 2. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 4-26**. **Figure 4-26** Clicking Create Access Key  3. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**. **NOTE** • Only two access keys can be added for each user. • To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.	-
SK		-

Parameter	Description	Example Value
Run Mode	This parameter is used only when the Hive version is **HIVE_3_X**. Possible values are: • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **Standalone**: The link instance runs in an independent process. If CDM needs to connect to multiple Hadoop data sources (MRS, Hadoop, or CloudTable) with both Kerberos and Simple authentication modes, **Standalone** prevails. **NOTE** The **STANDALONE** mode is used to solve the version conflict problem. If the connector versions of the source and destination ends of the same link are different, a JAR file conflict occurs. In this case, you need to place the source or destination end in the STANDALONE process to prevent the migration failure caused by the conflict.	EMBEDDED
Check Hive JDBC Connectivity	Whether to check the Hive JDBC connectivity	No
Use Cluster Config	You can use the cluster configuration to simplify parameter settings for the Hadoop connection.	No
Cluster Config Name	This parameter is valid only when **Use Cluster Config** is set to **Yes**. Select a cluster configuration that has been created. For details about how to configure a cluster, see **Managing Cluster Configurations**.	hive_01

Click **Show Advanced Attributes**, and then click **Add** to add configuration attributes of other clients. The name and value of each attribute must be configured. You can click **Delete** to delete no longer used attributes.

The following are some examples:

- **connectTimeout=360000** and **socketTimeout=360000**: When a large amount of data needs to be migrated or the entire table is retrieved using query statements, the migration fails due to connection timeout. In this case, you can customize the connection timeout interval (ms) and socket timeout interval (ms) to prevent failures caused by timeout.
- **hive.server2.idle.operation.timeout=360000**: To prevent Hive migration jobs from being suspended for a long time, you can customize the operation timeout period (ms).
- **hive.storeFormat=textfile**: During data migration from a relational database to Hive, tables in ORC format are automatically created by default. If you

want textfile or parquet tables to be created, add **hive.storeFormat=textfile** or **hive.storeFormat=parquet**.

- **fs.defaultFS=obs://hivedb**: If the interconnected MRS Hive uses decoupled storage and compute, you can use this configuration to achieve better compatibility.

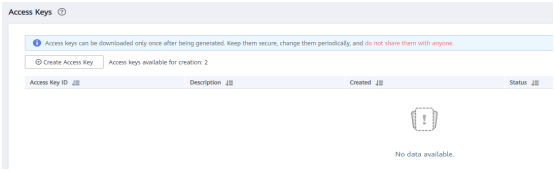
FusionInsight Hive

The FusionInsight Hive link is applicable to data migration of FusionInsight HD in the local data center. You must use Direct Connect to connect to FusionInsight HD.

Table 4-27 describes related parameters.

Table 4-27 FusionInsight Hive link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	hivelink
Manager IP	IP address of FusionInsight Manager	127.0.0.1
Manager Port	Port number of FusionInsight Manager	28443
CAS Server Port	Port number of the CAS server used to connect to FusionInsight	20009
Authentication Method	Authentication method used for accessing the cluster: • **SIMPLE**: Select this for non-security mode. • **KERBEROS**: Select this for security mode.	SIMPLE
HIVE Version	Hive version	HIVE_3_X
Username	Username used for logging in to FusionInsight Manager.	cdm
Password	Password used for logging in to FusionInsight Manager	-
OBS storage support	The server must support OBS storage. When creating a Hive table, you can store the table in OBS.	No

Parameter	Description	Example Value
AK	This parameter is mandatory when **OBS storage support** is enabled. The account corresponding to the AK/SK pair must have the OBS Buckets Viewer permission. Otherwise, OBS cannot be accessed and the "403 AccessDenied" error is reported. You need to create an access key for the current account and obtain an AK/SK pair. 1. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list. 2. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 4-27**. **Figure 4-27** Clicking Create Access Key  3. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**. **NOTE** • Only two access keys can be added for each user. • To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.	-
SK		-

Parameter	Description	Example Value
Run Mode	This parameter is used only when the Hive version is **HIVE_3_X**. Possible values are: • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **Standalone**: The link instance runs in an independent process. If CDM needs to connect to multiple Hadoop data sources (MRS, Hadoop, or CloudTable) with both Kerberos and Simple authentication modes, **Standalone** prevails. **NOTE** The **STANDALONE** mode is used to solve the version conflict problem. If the connector versions of the source and destination ends of the same link are different, a JAR file conflict occurs. In this case, you need to place the source or destination end in the STANDALONE process to prevent the migration failure caused by the conflict.	EMBEDDED
Use Cluster Config	You can use the cluster configuration to simplify parameter settings for the Hadoop connection.	No
Cluster Config Name	This parameter is valid only when **Use Cluster Config** is set to **Yes**. Select a cluster configuration that has been created. For details about how to configure a cluster, see **Managing Cluster Configurations**.	hive_01

Click **Show Advanced Attributes**, and then click **Add** to add configuration attributes of other clients. The name and value of each attribute must be configured. You can click **Delete** to delete no longer used attributes.

The following are some examples:

- **connectTimeout=360000** and **socketTimeout=360000**: When a large amount of data needs to be migrated or the entire table is retrieved using query statements, the migration fails due to connection timeout. In this case, you can customize the connection timeout interval (ms) and socket timeout interval (ms) to prevent failures caused by timeout.
- **hive.server2.idle.operation.timeout=360000**: To prevent Hive migration jobs from being suspended for a long time, you can customize the operation timeout period (ms).

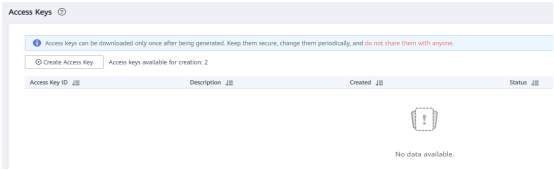
Apache Hive

The Apache Hive link is applicable to data migration of the third-party Hadoop in the local data center or ECS. You must use Direct Connect to connect to Hadoop in the local data center.

Table 4-28 describes related parameters.

Table 4-28 Apache Hive link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	hivelink
URI	NameNode URI	hdfs://hacluster
Hive Metastore	Hive metadata address. For details, see the **hive.metastore.uris** configuration item. Example: thrift://host-192-168-1-212:9083	-
Authentication Method	Authentication method used for accessing the cluster: ● **SIMPLE**: Select this for non-security mode. ● **KERBEROS**: Select this for security mode.	SIMPLE
Hive Version	Hive version	HIVE_3_X
IP and Host Name Mapping	If the Hadoop configuration file uses the host name, configure the mapping between the IP address and host name. Separate the IP addresses and host names by spaces and mappings by semicolons (;), carriage returns, or line feeds.	-
OBS storage support	The server must support OBS storage. When creating a Hive table, you can store the table in OBS.	No

Parameter	Description	Example Value
AK	This parameter is mandatory when **OBS storage support** is enabled. The account corresponding to the AK/SK pair must have the OBS Buckets Viewer permission. Otherwise, OBS cannot be accessed and the "403 AccessDenied" error is reported. You need to create an access key for the current account and obtain an AK/SK pair. 1. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list. 2. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 4-28**. **Figure 4-28** Clicking Create Access Key  3. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**. **NOTE** • Only two access keys can be added for each user. • To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.	-
SK		-

Parameter	Description	Example Value
Run Mode	This parameter is used only when the Hive version is **HIVE_3_X**. Possible values are: • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **Standalone**: The link instance runs in an independent process. If CDM needs to connect to multiple Hadoop data sources (MRS, Hadoop, or CloudTable) with both Kerberos and Simple authentication modes, **Standalone** prevails. **NOTE** The **STANDALONE** mode is used to solve the version conflict problem. If the connector versions of the source and destination ends of the same link are different, a JAR file conflict occurs. In this case, you need to place the source or destination end in the STANDALONE process to prevent the migration failure caused by the conflict.	EMBEDDED
Use Cluster Config	You can use the cluster configuration to simplify parameter settings for the Hadoop connection.	No
Cluster Config Name	This parameter is valid when **Use Cluster Config** is set to **Yes** or **Authentication Method** is set to **KERBEROS**. Select a cluster configuration that has been created. For details about how to configure a cluster, see **Managing Cluster Configurations**.	hive_01
Hive JDBC URL	URL for connecting to Hive JDBC. By default, anonymous users are used.	-

Click **Show Advanced Attributes**, and then click **Add** to add configuration attributes of other clients. The name and value of each attribute must be configured. You can click **Delete** to delete no longer used attributes.

The following are some examples:

- **connectTimeout=360000** and **socketTimeout=360000**: When a large amount of data needs to be migrated or the entire table is retrieved using query statements, the migration fails due to connection timeout. In this case, you can customize the connection timeout interval (ms) and socket timeout interval (ms) to prevent failures caused by timeout.
- **hive.server2.idle.operation.timeout=360000**: To prevent Hive migration jobs from being suspended for a long time, you can customize the operation timeout period (ms).

4.5.11 Link to HBase

CDM supports the following HBase data sources:

- **MRS HBase**
- **FusionInsight HBase**
- **Apache HBase**

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

MRS HBase

When connecting CDM to HBase of MRS, configure the parameters as described in **Table 4-29**.

NOTE

- Before creating an MRS link, you need to add an authenticated Kerberos user on MRS and log in to the MRS management page to change the initial password. Then use the new user to create an MRS link.
- To connect to an MRS 2.*x* cluster, create a CDM cluster of version 2.*x* first. CDM 1.8.*x* clusters cannot connect to MRS 2.*x* clusters.
- If the connection fails after you select a cluster, check whether the MRS cluster can communicate with the CDM instance which functions as the agent. They can communicate with each other in the following scenarios:
  - If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule.
  - If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**.
  - The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.

Table 4-29 MRS HBase link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	mrs_hbase_link
Manager IP	Floating IP address of MRS Manager. Click **Select** next to the **Manager IP** text box to select an MRS cluster. CDM automatically fills in the authentication information.	127.0.0.1

Parameter	Description	Example Value
Username	If **Authentication Method** is set to **KERBEROS**, you must provide the username and password used for logging in to MRS Manager. If you need to create a snapshot when exporting a directory from HDFS, the user configured here must have the administrator permission on HDFS. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user and set **Username** and **Password** to the username and password of the created MRS user when creating an MRS data connection. **NOTE** • If the CDM cluster version is 2.9.0 or later and the MRS cluster version is 3.1.0 or later, the created user must have the permissions of the **Manager_viewer** role to create links on CDM. To perform operations on databases, tables, and columns of an MRS component, you also need to add the database, table, and column permissions of the MRS component to the user by following the instructions in the MRS documentation. • If the CDM cluster version is earlier than 2.9.0 or the MRS cluster version is earlier than 3.1.0, the created user must have the permissions of **Manager_administrator** or **System_administrator** to create links on CDM. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.	cdm
Password	Password used for logging in to MRS Manager	-
Authentication Method	Authentication method used for accessing the cluster: • **SIMPLE**: Select this for non-security mode. • **KERBEROS**: Select this for security mode.	SIMPLE
HBase Version	HBase version	HBASE_2_X

Parameter	Description	Example Value
Run Mode	Run mode of the HBase link. This parameter is used only when the HBase version is **HBASE_2_X**. • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **Standalone**: The link instance runs in an independent process. If CDM needs to connect to multiple Hadoop data sources (MRS, Hadoop, or CloudTable) with both Kerberos and Simple authentication modes, **Standalone** prevails. Note: The STANDALONE mode is used to solve the version conflict problem. If the connector versions of the source and destination ends of the same link are different, a JAR file conflict occurs. In this case, you need to place the source or destination end in the STANDALONE process to prevent the migration failure caused by the conflict.	STANDALONE
Use Cluster Config	You can create cluster configurations on the **Links** page to simplify the configuration of Hadoop link parameters.	No
Cluster Config Name	This parameter is valid only when **Use Cluster Config** is set to **Yes**. Select a cluster configuration that has been created. For details about how to configure a cluster, see **Managing Cluster Configurations**.	hbase_01

Click **Show Advanced Attributes**, and then click **Add** to add configuration attributes of other clients. The name and value of each attribute must be configured. You can click **Delete** to delete no longer used attributes.

FusionInsight HBase

When connecting CDM to HBase of FusionInsight HD, configure the parameters as described in **Table 4-30**.

Table 4-30 FusionInsight HBase link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	FI_hbase_link
Manager IP	IP address of FusionInsight Manager	127.0.0.1
Manager Port	Port number of FusionInsight Manager	28443
CAS Server Port	Port number of the CAS server used to connect to FusionInsight	20009
Username	Username used for logging in to FusionInsight Manager.	cdm
Password	Password used for logging in to FusionInsight Manager	-
Authentication Method	Authentication method used for accessing the cluster: • **SIMPLE**: Select this for non-security mode. • **KERBEROS**: Select this for security mode.	Kerberos
HBase Version	HBase version	HBASE_2_X
Run Mode	Run mode of the HBase link. This parameter is used only when the HBase version is **HBASE_2_X**. • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **Standalone**: The link instance runs in an independent process. If CDM needs to connect to multiple Hadoop data sources (MRS, Hadoop, or CloudTable) with both Kerberos and Simple authentication modes, **Standalone** prevails. **NOTE** The **STANDALONE** mode is used to solve the version conflict problem. If the connector versions of the source and destination ends of the same link are different, a JAR file conflict occurs. In this case, you need to place the source or destination end in the STANDALONE process to prevent the migration failure caused by the conflict.	STANDALONE
Use Cluster Config	You can use the cluster configuration to simplify parameter settings for the Hadoop connection.	No

Parameter	Description	Example Value
Cluster Config Name	This parameter is valid only when **Use Cluster Config** is set to **Yes**. Select a cluster configuration that has been created. For details about how to configure a cluster, see **Managing Cluster Configurations**.	hbase_01

Click **Show Advanced Attributes**, and then click **Add** to add configuration attributes of other clients. The name and value of each attribute must be configured. You can click **Delete** to delete no longer used attributes.

Apache HBase

When connecting CDM to HBase of Apache Hadoop, configure the parameters as described in **Table 4-31**.

Table 4-31 Apache HBase link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	hadoop_hbase_link
ZK Link	ZooKeeper link of HBase Format: <host1>:<port>,<host2>:<port>,<host3>:<port>	zk1.example.com:2181,zk2.example.com:2181,zk3.example.com:2181
Authentication Method	Authentication method used for accessing the cluster: • **SIMPLE**: Select this for non-security mode. • **KERBEROS**: Select this for security mode.	Kerberos
IP and Host Name Mapping	IP address and host name. If the configuration file uses host names, configure the mappings between all IP addresses and hosts. Use spaces to separate hosts.	IP: 10.3.6.9 Host name: hostname01
HBase Version	HBase version	HBASE_2_X

Parameter	Description	Example Value
Run Mode	Run mode of the HBase link. This parameter is used only when the HBase version is **HBASE_2_X**. • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **Standalone**: The link instance runs in an independent process. If CDM needs to connect to multiple Hadoop data sources (MRS, Hadoop, or CloudTable) with both Kerberos and Simple authentication modes, **Standalone** prevails. **NOTE** The **STANDALONE** mode is used to solve the version conflict problem. If the connector versions of the source and destination ends of the same link are different, a JAR file conflict occurs. In this case, you need to place the source or destination end in the STANDALONE process to prevent the migration failure caused by the conflict.	STANDALONE
Use Cluster Config	You can use the cluster configuration to simplify parameter settings for the Hadoop connection.	No
Cluster Config Name	This parameter is valid only when **Use Cluster Config** is set to **Yes**. Select a cluster configuration that has been created. For details about how to configure a cluster, see **Managing Cluster Configurations**.	hbase_01

Click **Show Advanced Attributes**, and then click **Add** to add configuration attributes of other clients. The name and value of each attribute must be configured. You can click **Delete** to delete no longer used attributes.

4.5.12 Link to HDFS

CDM supports the following HDFS data sources:

- **MRS HDFS**
- **FusionInsight HDFS**
- **Apache HDFS**

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

MRS HDFS

When connecting CDM to HDFS of MRS, configure the parameters as described in **Table 4-32**.

NOTE

- Before creating an MRS link, you need to add an authenticated Kerberos user on MRS and log in to the MRS management page to change the initial password. Then use the new user to create an MRS link.
- To connect to an MRS 2.*x* cluster, create a CDM cluster of version 2.*x* first. CDM 1.8.*x* clusters cannot connect to MRS 2.*x* clusters.
- If the connection fails after you select a cluster, check whether the MRS cluster can communicate with the CDM instance which functions as the agent. They can communicate with each other in the following scenarios:
  - If the CDM cluster in the DataArts Studio instance and the MRS cluster are in different regions, a public network or a dedicated connection is required. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, and the MRS cluster can access the Internet and the port has been enabled in the firewall rule.
  - If the CDM cluster in the DataArts Studio instance and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**.
  - The MRS cluster and the DataArts Studio workspace belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.

Table 4-32 MRS HDFS link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	mrs_hdfs_link
Manager IP	Floating IP address of MRS Manager. Click **Select** next to the **Manager IP** text box to select an MRS cluster. CDM automatically fills in the authentication information.	127.0.0.1

Parameter	Description	Example Value
Username	If **Authentication Method** is set to **KERBEROS**, you must provide the username and password used for logging in to MRS Manager. If you need to create a snapshot when exporting a directory from HDFS, the user configured here must have the administrator permission on HDFS. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user and set **Username** and **Password** to the username and password of the created MRS user when creating an MRS data connection. **NOTE** ● If the CDM cluster version is 2.9.0 or later and the MRS cluster version is 3.1.0 or later, the created user must have the permissions of the **Manager_viewer** role to create links on CDM. To perform operations on databases, tables, and columns of an MRS component, you also need to add the database, table, and column permissions of the MRS component to the user by following the instructions in the MRS documentation. ● If the CDM cluster version is earlier than 2.9.0 or the MRS cluster version is earlier than 3.1.0, the created user must have the permissions of **Manager_administrator** or **System_administrator** to create links on CDM. ● A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.	cdm
Password	Password used for logging in to MRS Manager	-
Authentication Method	Authentication method used for accessing MRS ● **SIMPLE**: Select this for non-security mode. ● **KERBEROS**: Select this for security mode.	SIMPLE

Parameter	Description	Example Value
Run Mode	Run mode of the HDFS link. The options are as follows: • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **STANDALONE**: The link instance runs in an independent process. If you want to connect CDM to multiple Hadoop data sources (MRS, Hadoop, or CloudTable), and both **KERBEROS** and **SIMPLE** authentication modes are available, you must select **STANDALONE** for this parameter. Note: The STANDALONE mode is used to solve the version conflict problem. If the connector versions of the source and destination ends of the same link are different, a JAR file conflict occurs. In this case, you need to place the source or destination end in the STANDALONE process to prevent the migration failure caused by the conflict. If a CDM cluster connects to two or more clusters with Kerberos authentication enabled and the same realm, only one cluster can be connected in **EMBEDDED** mode, and the other clusters must be connected in **STANDALONE** mode.	STANDALONE
Agent	The agent function will be unavailable soon and does not need to be configured.	-
Use Cluster Config	You can use the cluster configuration to simplify parameter settings for the Hadoop connection.	No
Cluster Config Name	This parameter is valid only when **Use Cluster Config** is set to **Yes**. Select a cluster configuration that has been created. For details about how to configure a cluster, see **Managing Cluster Configurations**.	hdfs_01

Click **Show Advanced Attributes**, and then click **Add** to add configuration attributes of other clients. The name and value of each attribute must be configured. You can click **Delete** to delete no longer used attributes.

FusionInsight HDFS

When connecting CDM to HDFS of FusionInsight HD, configure the parameters as described in **Table 4-33**.

Table 4-33 FusionInsight HDFS link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	FI_hdfs_link
Manager IP	IP address of FusionInsight Manager	127.0.0.1
Manager Port	Port number of FusionInsight Manager	28443
CAS Server Port	Port number of the CAS server used to connect to FusionInsight	20009
Username	Username used for logging in to FusionInsight Manager. If you need to create a snapshot when exporting a directory from HDFS, the user configured here must have the administrator permission on HDFS.	cdm
Password	Password used for logging in to FusionInsight Manager	-
Authentication Method	Authentication method used for accessing the cluster: • **SIMPLE**: Select this for non-security mode. • **KERBEROS**: Select this for security mode.	KERBEROS

Parameter	Description	Example Value
Run Mode	Run mode of the HDFS link. The options are as follows: • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **STANDALONE**: The link instance runs in an independent process. If you want to connect CDM to multiple Hadoop data sources (MRS, Hadoop, or CloudTable), and both **KERBEROS** and **SIMPLE** authentication modes are available, you must select **STANDALONE** for this parameter. Note: The STANDALONE mode is used to solve the version conflict problem. If the connector versions of the source and destination ends of the same link are different, a JAR file conflict occurs. In this case, you need to place the source or destination end in the STANDALONE process to prevent the migration failure caused by the conflict.	STANDALONE
Agent	The agent function will be unavailable soon and does not need to be configured.	-
Use Cluster Config	You can use the cluster configuration to simplify parameter settings for the Hadoop connection.	No
Cluster Config Name	This parameter is valid only when **Use Cluster Config** is set to **Yes**. Select a cluster configuration that has been created. For details about how to configure a cluster, see **Managing Cluster Configurations**.	hdfs_01

Click **Show Advanced Attributes**, and then click **Add** to add configuration attributes of other clients. The name and value of each attribute must be configured. You can click **Delete** to delete no longer used attributes.

Apache HDFS

When connecting CDM to HDFS of Apache Hadoop, configure the parameters as described in **Table 4-34**.

Table 4-34 Apache HDFS link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	hadoop_hdfs_link
URI	NameNode URI You can enter **hdfs://*IP address of the NameNode instance*:8020**.	hdfs://*IP*:8020
Authentication Method	Authentication method used for accessing the cluster: ● **SIMPLE**: Select this for non-security mode. ● **KERBEROS**: Select this for security mode.	KERBEROS
Run Mode	Run mode of the HDFS link. The options are as follows: ● **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. ● **STANDALONE**: The link instance runs in an independent process. If you want to connect CDM to multiple Hadoop data sources (MRS, Hadoop, or CloudTable), and both **KERBEROS** and **SIMPLE** authentication modes are available, you must select **STANDALONE** for this parameter. Note: The STANDALONE mode is used to solve the version conflict problem. If the connector versions of the source and destination ends of the same link are different, a JAR file conflict occurs. In this case, you need to place the source or destination end in the STANDALONE process to prevent the migration failure caused by the conflict.	STANDALONE
IP and Host Name Mapping	This parameter is used only when **Run Mode** is set to **EMBEDDED** or **STANDALONE**. If the HDFS configuration file uses the host name, configure the mapping between the IP address and host name. Separate the IP addresses and host names by spaces and mappings by semicolons (;), carriage returns, or line feeds.	10.1.6.9 hostname01 10.2.7.9 hostname02
Agent	The agent function will be unavailable soon and does not need to be configured.	-

Parameter	Description	Example Value
Use Cluster Config	You can use the cluster configuration to simplify parameter settings for the Hadoop connection.	No
Cluster Config Name	This parameter is valid when **Use Cluster Config** is set to **Yes** or **Authentication Method** is set to **KERBEROS**. Select a cluster configuration that has been created. For details about how to configure a cluster, see **Managing Cluster Configurations**.	hdfs_01

4.5.13 Link to an FTP or SFTP Server

The FTP/SFTP link is used to migrate files from the on-premises file server or ECS to a database.

NOTE

- Only FTP servers running Linux are supported.
- Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

When connecting CDM to an FTP or SFTP server, configure the parameters as described in **Table 4-35**.

Table 4-35 FTP/SFTP link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	ftp_link
Host Name/IP Address	Host name or IP address of the FTP or SFTP server	ftp.apache.org
Port	Port number of the FTP or SFTP server. The default value is **21** for FTP and **22** for SFTP.	21
Username	Username used for logging in to the FTP or SFTP server	cdm
Password	Password used for logging in to the FTP or SFTP server	-

4.5.14 Link to Redis

The Redis link is applicable to data migration of Redis created in the local data center or ECS. It is used to load data in the database or files to Redis.

Links to Redis data encrypted using SSL are not supported.

When connecting CDM to an on-premises Redis database, configure the parameters as described in **Table 4-36**.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-36 Redis link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	redis_link
Redis Deployment Method	Two deployment methods are available: • **Single**: installation on a single-node system • **Cluster**: installation on a cluster • **Proxy**: installation using a proxy	Single
Redis Server List	List of Redis server addresses. Enter each address in the format of *IP address or domain name of the database server*:*port number*, Separate multiple server lists by semicolons (;).	192.168.0.1:7300;192.168.0.2:7301
Password	Password used for logging in to Redis	-
Redis Database Index	Index ID of a Redis database A Redis database is similar to a relational database. The total number of Redis databases can be set in the Redis configuration file. By default, there are 16 Redis databases. The database names are integers ranging from 0 to 15 instead of character strings.	0
Authentication Method	Authentication method used for accessing MRS • **SIMPLE**: Select this for non-security mode. • **KERBEROS**: Select this for security mode.	SIMPLE

Parameter	Description	Example Value
Username	If **Authentication Method** is set to **KERBEROS**, you must provide the username and password used for logging in to MRS Manager. If you need to create a snapshot when exporting a directory from HDFS, the user configured here must have the administrator permission on HDFS. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user and set **Username** and **Password** to the username and password of the created MRS user when creating an MRS data connection. **NOTE** • If the CDM cluster version is 2.9.0 or later and the MRS cluster version is 3.1.0 or later, the created user must have the permissions of the **Manager_viewer** role to create links on CDM. To perform operations on databases, tables, and columns of an MRS component, you also need to add the database, table, and column permissions of the MRS component to the user by following the instructions in the MRS documentation. • If the CDM cluster version is earlier than 2.9.0 or the MRS cluster version is earlier than 3.1.0, the created user must have the permissions of **Manager_administrator** or **System_administrator** to create links on CDM. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.	cdm
Cluster Config Name	This parameter is valid only when **Authentication Method** is set to **KERBEROS**. Select a cluster configuration you have created. For details about how to configure a cluster, see **Managing Cluster Configurations**.	hdfs_01

4.5.15 Link to DDS

The DDS link is used to synchronize data from Document Database Service (DDS) on HUAWEI CLOUD to a big data platform.

When connecting CDM to DDS, configure the parameters as described in **Table 4-37**.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-37 DDS link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	dds_link
Server List	List of server addresses. Enter each address in the format of *IP address or domain name of the database server*:*port number*, and separate the entered addresses with semicolons (;).	192.168.0.1:7300;192.168.0.2:7301
Database Name	Name of the DDS database to be connected	DB_dds
Username	Username used for logging in to DDS	cdm
Password	Password used for logging in to DDS	-
Is direct connection mode	This mode applies to the scenario where the network of the primary node is normal but that of the replica node is abnormal. **NOTE** • Only one IP address can be configured for the server list in direct connection mode. • This mode applies to the scenario where the network of the primary node is normal but the network of the replica node is abnormal.	No

4.5.16 Link to CloudTable

When connecting CDM to CloudTable, configure the parameters as described in **Table 4-38**.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-38 CloudTable link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	cloudtable_link
ZK Link	Obtain this parameter value from the cluster management page of CloudTable.	cloudtable-cdm-zk1.cloudtable.com:2181,cloudtable-cdm-zk2.cloudtable.com:2181
IAM Authentication	If IAM authentication is enabled for the CloudTable cluster to be connected, set this parameter to **Yes**. Otherwise, set this to **No**. If you select **Yes**, enter the username, AK, and SK.	No
Username	Username used for accessing the CloudTable cluster	admin
AK	AK for accessing the CloudTable cluster. You need to create an access key for the current account and obtain an AK/SK pair.	-
SK	SK for accessing the CloudTable cluster. You need to create an access key for the current account and obtain an AK/SK pair.	-
Use Cluster Config	You can use the cluster configuration to simplify parameter settings for the Hadoop connection.	No
Cluster Config Name	This parameter is valid only when **Use Cluster Config** is set to **Yes**. Select a cluster configuration that has been created. For details about how to configure a cluster, see **Managing Cluster Configurations**.	hadoop_01

Click **Show Advanced Attributes**, and then click **Add** to add configuration attributes of other clients. The name and value of each attribute must be configured. You can click **Delete** to delete no longer used attributes.

4.5.17 Link to MongoDB

This link is used to transfer data from a third-party cloud MongoDB service or MongoDB created in the on-premises data center or ECS to a big data platform.

When connecting CDM to an on-premises MongoDB database, configure the parameters as described in **Table 4-39**.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-39 MongoDB link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	mongodb_link
Server List	List of MongoDB server addresses. Enter each address in the format of *IP address or domain name of the database server*:*port number*, and separate the entered addresses with semicolons (;).	192.168.0.1:7300;192.168.0.2:7301
Database Name	Name of the MongoDB database to be connected	DB_mongodb
Username	Username for logging in to MongoDB	cdm
Password	Password for logging in to MongoDB	-
Direct Connection	This mode applies to the scenario where the network of the primary node is normal but the network of the replica node is abnormal. **NOTE** • Only one IP address can be configured for the server list in direct connection mode. • This mode applies to the scenario where the network of the primary node is normal but the network of the replica node is abnormal.	No
Link Attributes	Custom link attributes. The MongoDB attributes are supported. The unit is ms. The link attributes are as follows: • **socketTimeout**: The default value is **60000**. • **maxWaitTime**: The default value is **10000**. • **connectTimeout**. The default value is **10000**. • **serverSelectionTimeout**: The default value is **5000**.	socketTimeout=60000

4.5.18 Link to Cassandra

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-40 Parameter description

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	mongodb_link
Service node	An address of one node or addresses of multiple nodes. Separate addresses with semicolons (;). You are advised to configure multiple nodes at a time.	192.168.0.1;192.168.0.2
Port	Port number of the Cassandra node to be connected.	9042
Username	User name for connecting to Cassandra.	cdm
Password	Password for connecting to Cassandra.	-
Connection timeout duration	(Optional) Displayed when you click **Show Advanced Attributes**. Connection timeout interval, in seconds.	5
Read timeout duration	(Optional) Displayed when you click **Show Advanced Attributes**. Read timeout interval, in seconds. If the value is less than or equal to 0, no timeout occurs.	12

4.5.19 Link to DIS

When connecting CDM to DIS, configure the parameters as described in **Table 4-41**.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-41 DIS link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	dis_link
Region	Region where DIS is deployed	-
Endpoint	URL of DIS in the format of https://*Endpoint*. An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions. You can obtain endpoints of the service from **Endpoints**.	-
AK	AK used for logging in to the DIS server. You need to create an access key for the current account and obtain an AK/SK pair.	-
SK	SK used for logging in to the DIS server. You need to create an access key for the current account and obtain an AK/SK pair.	-
Project ID	Project ID of DIS	-

4.5.20 Link to Kafka

MRS Kafka

When connecting CDM to Kafka of MRS, configure the parameters as described in **Table 4-42**.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-42 MRS Kafka link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	kafka_link
Manager IP	Floating IP address of MRS Manager. Click **Select** next to the **Manager IP** text box to select an MRS cluster. CDM automatically fills in the authentication information.	127.0.0.1

Parameter	Description	Example Value
Username	Username used for logging in to MRS Manager To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user and set **Username** and **Password** to the username and password of the created MRS user when creating an MRS data connection. **NOTE** • If the CDM cluster version is 2.9.0 or later and the MRS cluster version is 3.1.0 or later, the created user must have the permissions of the **Manager_viewer** role to create links on CDM. To perform operations on databases, tables, and columns of an MRS component, you also need to add the database, table, and column permissions of the MRS component to the user by following the instructions in the MRS documentation. • If the CDM cluster version is earlier than 2.9.0 or the MRS cluster version is earlier than 3.1.0, the created user must have the permissions of **Manager_administrator** or **System_administrator** to create links on CDM. • A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.	-
Password	Password used for logging in to MRS Manager	-
Authentication Method	Authentication method used for accessing MRS • **SIMPLE**: for non-security mode • **KERBEROS**: for security mode	Yes

Click **Show Advanced Attributes**, and then click **Add** to add configuration attributes of other clients. The name and value of each attribute must be configured. You can click **Delete** to delete no longer used attributes.

Apache Kafka

The Apache Kafka link is applicable to data migration of the third-party Hadoop in the local data center or ECS. You must use Direct Connect to connect to Hadoop in the local data center.

When connecting CDM to Kafka of Apache Hadoop, configure the parameters as described in **Table 4-43**.

Table 4-43 Parameter description

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	kafka_link
Kafka broker	IP address and port number of the Kafka broker	192.168.1.1:9092

Click **Show Advanced Attributes**, and then click **Add** to add configuration attributes of other clients. The name and value of each attribute must be configured. You can click **Delete** to delete no longer used attributes.

4.5.21 Link to DMS Kafka

When connecting CDM to DMS Kafka, configure the parameters as described in **Table 4-44**.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-44 DMS Kafka link parameter

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	dms_link
Service Type	DMS Kafka edition. Currently, only the Platinum edition is available.	Platinum
Kafka Broker	Address of a Kafka premium instance. The format is host:port.	-

Parameter	Description	Example Value
Kafka SASL_SSL	Whether to enable SSL authentication when a client connects to a Kafka premium instance. This function must be enabled if the SASL_SSL security protocol is enabled for the link to the DMS Kafka instance. If Kafka SASL_SSL is enabled, data will be encrypted before transmission for higher security, but performance will suffer. **NOTE** When SSL authentication is enabled, Kafka continuously parses the Kafka broker connection address as a domain name, which undermines performance. You are advised to add the self-mapping of the broker connection address to the **/etc/hosts** file on the ECS corresponding to the CDM cluster (search for the ECS based on the cluster IP address) so that the client can quickly resolve the broker of the instance. For example, if the Kafka broker address is 10.154.48.120, add the following self-mapping to the **/etc/hosts** file: 10.154.48.120 10.154.48.120	Yes
Username	Username for connecting to DMS Kafka. This parameter is displayed when **Kafka SASL_SSL** is enabled.	-
Password	Password for connecting to DMS Kafka. This parameter is displayed when **Kafka SASL_SSL** is enabled.	-
Kafka Properties	• If a security protocol is enabled for the link to the DMS Kafka instance, you must add a data encryption attribute, and set the attribute name to **security.protocol** and value to **SASL_SSL** or **SASL_PLAINTEXT** based on the security protocol of the Kafka instance. • If SASL authentication is enabled for the link to the DMS Kafka instance, you must add an authentication mode attribute, and set the attribute name to **sasl.mechanism** and value to **PLAIN** or **SCRAM-SHA-512** based on the SASL authentication mechanism configured for the Kafka instance (set the value to either **PLAIN** or **SCRAM-SHA-512** if both are supported).	-

4.5.22 Link to CSS

Huawei Cloud Cloud Search Service (CSS) is a fully hosted distributed search service powered by open-source Elasticsearch. CSS links can be used to migrate log files and database records to CSS for search and analysis using Elasticsearch.

NOTE

- You are advised to use Logstash to import data to CSS. For details, see **Using Logstash to Import Data to Elasticsearch**.
- Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-45 lists the parameters for a CSS link.

Table 4-45 CSS link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	css_link
Elasticsearch Server List	IP addresses or domain names (including the port numbers) of one or more Elasticsearch servers. The format is *ip:port*. Use semicolons (;) to separate multiple IP addresses.	192.168.0.1:9200;192.168.0.2:9200
Security Mode Authentication	Whether to enable security mode. If **Security Mode** has been enabled for the CSS cluster to be connected, set this parameter to **Yes**. Otherwise, set this to **No**.	Yes
Username	This parameter is displayed when **Security Mode Authentication** is set to **Yes**. It indicates the username used for connecting to CSS.	admin
Password	This parameter is displayed when **Security Mode Authentication** is set to **Yes**. It indicates the password used for connecting to CSS.	-
HTTPS Access	This parameter is displayed when **Security Mode Authentication** is set to **Yes**. This parameter specifies whether to enable HTTPS access. HTTPS access is more secure than HTTP access.	Yes

4.5.23 Link to Elasticsearch

Elasticsearch links can be used to connect to Elasticsearch services in third-party clouds and local data centers and on Elastic Cloud Servers (ECSs).

NOTE

- The Elasticsearch connector only supports Elasticsearch clusters in non-security mode.
- Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-46 lists the parameters for an Elasticsearch link.

Table 4-46 Elasticsearch link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	es_link
Elasticsearch Server List	IP addresses or domain names (including the port numbers) of one or more Elasticsearch servers. The format is *ip:port*. Use semicolons (;) to separate multiple IP addresses or domain names.	192.168.0.1:9200;192.168.0.2:9200

4.5.24 Link to a Dameng Database

When connecting CDM to a Dameng database, configure the parameters as described in Table 4-47.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-47 Parameters for a link to a Dameng database

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	dm_link
Database Server	IP address or domain name of the database to connect Click **Select** next to the text box and select a DWS or RDS DB instance in the displayed dialog box.	192.168.0.1

Parameter	Description	Example Value
Port	Port of the database to connect	The port number varies depending on the database.
Database Name	Name of the database to connect	dbname
Username	Username used for accessing the database This account must have the permissions required to read and write data tables and metadata.	cdm
Password	Password of the user	-
Driver Version	Select a driver version that adapts to the database type.	-
Fetch Size	(Optional) Displayed when you click **Show Advanced Attributes**. Number of rows obtained by each request. Set this parameter based on the data source and the job's data size. If the value is either too large or too small, the job may run for a long time.	1000
Link Attributes	(Optional) Click **Add** to add the JDBC connector attributes of multiple specified data sources. For details, see the JDBC connector document of the corresponding database.	sslmode=require
Reference Sign	(Optional) Delimiter between the names of the referenced tables or columns. For details, see the product documentation of the corresponding database.	'

4.5.25 Link to SAP HANA

Table 4-48 describes the SAP HANA link parameters.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-48 SAP HANA link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	sap_link

Parameter	Description	Example Value
Database Server	IP address or domain name of the database to connect Click **Select** next to the text box to obtain the list of instances.	192.168.0.1
Port	Port of the database to connect	The port number varies depending on the database.
Database Name	Name of the database to connect	dbname
Username	Username used for accessing the database. This user must have the permissions to read and write data tables and metadata.	cdm
Password	Password of the user	-
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-
Agent	The agent function will be unavailable soon and does not need to be configured.	-
Fetch Size	(Optional) Displayed when you click **Show Advanced Attributes**. Number of rows obtained by each request. Set this parameter based on the data source and the job's data size. If the value is either too large or too small, the job may run for a long time.	1000

Parameter	Description	Example Value
Link Attributes	(Optional) Click **Add** to add the JDBC connector attributes of multiple specified data sources. For details, see the JDBC connector document of the corresponding database. The following are some examples: • **connectTimeout=360000** and **socketTimeout=360000**: When a large amount of data needs to be migrated or the entire table is retrieved using query statements, the migration fails due to connection timeout. In this case, you can customize the connection timeout interval (ms) and socket timeout interval (ms) to prevent failures caused by timeout. • **useCursorFetch=false**: By default, **useCursorFetch** is enabled, indicating that the JDBC connector communicates with relational databases using a binary protocol. Some third-party systems may have compatibility issues, causing migration time conversion errors. In this case, you can disable this function. Open-source MySQL databases support the **useCursorFetch** parameter, and you do not need to set this parameter.	sslmode=require
Reference Sign	(Optional) Delimiter between the names of the referenced tables or columns. For details, see the product documentation of the corresponding database.	'

4.5.26 Link to a Database Shard

Sharding refers to the link to multiple backend data sources at the same time. The link can be used as the job source to migrate data from multiple data sources to other data sources. **Table 4-49** lists the link parameters.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-49 Database shard link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	my_link
Username	Username used for accessing the database For a backend database A, this configuration takes effect only when no username and password are configured for A in the data source list. For a backend database B that has configured the username and password, this configuration does not takes effect.	cdm
Password	Password used for accessing the database. For a backend database A, this configuration takes effect only when no username and password are configured for A in the data source list. For a backend database B that has configured the username and password, this configuration does not takes effect.	-
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-
Agent	The agent function will be unavailable soon and does not need to be configured.	-
backendDatasource	Enter the type of the backend database. Currently, only MySQL is supported.	MySQL
Data Source List	Enter the IP address, port number, database name, account name, and password of the backend database, and separate them with colons (:). That is, ip:port:dbs:username:password. You can leave username:password empty. In this case, the username and password are used. If there are multiple backend databases, ensure that the table structures are the same and use vertical bars (|) to separate data sources. If the password contains a vertical bar (|) or colon (:), use a backslash (\) to escape the vertical bar. For example, **192.168.3.0:3306:cdm| 192.168.2.2:3306:cdm:user:password** indicates that the IP address of the first backend database is **192.168.3.0**, the port number is **3306**, the database name is **cdm**, and the account name and password are configured in *user* and *password*. The IP address of the second backend database is **192.168.2.2**, the port number is **3306**, the database name is **cdm**, the account name is **user** and the password is **password**.	192.168.3.0:3306:cdm| 192.168.2.2:3306:cdm:user:password

Parameter	Description	Example Value
Fetch Size	(Optional) Displayed when you click **Show Advanced Attributes**. Number of rows obtained by each request. Set this parameter based on the data source and the job's data size. If the value is either too large or too small, the job may run for a long time.	1000
Link Attributes	(Optional) Click **Add** to add the JDBC connector attributes of multiple specified data sources. For details, see the JDBC connector document of the corresponding database.	sslmode=require
Reference Sign	(Optional) Delimiter between the names of the referenced tables or columns. For details, see the product documentation of the corresponding database.	'

4.5.27 Link to MRS Hudi

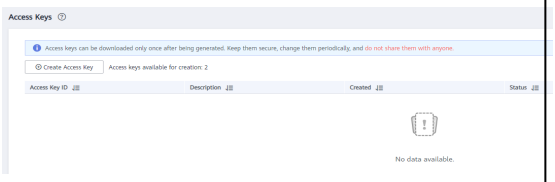
Table 4-50 describes the MRS Hudi link parameters.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-50 Hudi link parameters

Parameter	Description	Example Value
Name	Link name	Hudilink
Manager IP	Floating IP address of MRS Manager. Click **Select** next to the **Manager IP** text box to select an MRS cluster. CDM automatically fills in the authentication information.	127.0.0.1
Authentication Method	Authentication method used for accessing MRS • **SIMPLE**: Select this for non-security mode. • **KERBEROS**: Select this for security mode.	KERBEROS
Account	Username for logging in to MRS Manager	cdm
Password	Password for logging in to MRS Manager	-

Parameter	Description	Example Value
OBS storage support	Whether to support OBS storage. If the Hudi table data is stored in OBS, you need to enable this function.	Yes
AK	This parameter is available when **OBS storage support** is set to **Yes**. AK and SK are used to log in to the OBS server. You need to create an access key for the current account and obtain an AK/SK pair. To obtain an access key, perform the following steps: 1. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list. 2. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 4-29**. **Figure 4-29** Clicking Create Access Key  3. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**. **NOTE** • Only two access keys can be added for each user. • To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.	-
SK		-

Parameter	Description	Example Value
OBS Test Path	This parameter is available when **OBS storage support** is set to **Yes**. Enter a complete file path. The permission to access the path will be verified through the metadata query API. **NOTE** • For object storage, the path must be accurate to object, for example, **obs://bucket/dir/test.txt**. Otherwise, a 404 error occurs. • For a parallel file system, the path must be accurate to directory, for example, **obs://bucket/dir**.	obs://bucket/dir/test.txt
Hive Properties	Names of the tables to be integrated. Use commas (,) to separate multiple table names. This parameter is mandatory and cannot contain spaces.	-

4.5.28 Link to MRS ClickHouse

Table 4-51 describes the MRS ClickHouse link parameters.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-51 ClickHouse link parameters

Parameter	Description	Example Value
Name	Link name	cklink
Database Server	IP address or domain name of the database to connect Log in to Manager of the cluster where the MRS ClickHouse data source is located, choose **Cluster** > **Services** > **ClickHouse** > **Instance**, and view the ClickHouseServer service IP address.	192.168.0.1

Parameter	Description	Example Value
Port	Port of the database to connect **NOTE** • If the Server node is used, enable **SSL Encryption** and set the default port. Log in to the Manager of the cluster where the MRS ClickHouse data source is located, choose **Cluster** > **Services** > **ClickHouse** > **Instance**, and set the default port of ClickHouseServer. For an MRS cluster in non-security mode, set it to the value of the **http_port** parameter. For an MRS cluster in security mode, set it to the value of the **https_port** parameter. • If the Balancer node is used, enable **SSL Encryption** and set the default port. Log in to the Manager of the cluster where the MRS ClickHouse data source is located, choose **Cluster** > **Services** > **ClickHouse** > **Instance**, and set the default port of ClickHouseBalancer. For an MRS cluster in non-security mode, set it to the value of the **lb_http_port** parameter. For an MRS cluster in security mode, set it to the value of the **lb_https_port** parameter. • If MRS ClickHouse is deployed in a security cluster, set this parameter to the default HTTPS port.	8123
Database Name	Name of the database to connect	dbname
Username	Username used for accessing the database. This user must have the permissions to read and write data tables and metadata.	cdm
Password	Password of the user	-
SSL Encryption	(Optional) If you set this parameter to **Yes**, CDM can connect to the database (on-premises databases excluded) in SSL encryption mode.	No
Reference Sign	(Optional) Delimiter between the names of the referenced tables or columns. For details, see the product documentation of the corresponding database.	'

4.5.29 Link to a ShenTong Database

Table 4-52 lists the parameters for a link to a ShenTong database.

NOTE

Do not change the password or user when the job is running. If you do so, the password will not take effect immediately and the job will fail.

Table 4-52 ShenTong database link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	st_link
Database Server	IP address or domain name of the database to connect Click **Select** next to the text box and select a ShenTong DB instance in the displayed dialog box.	192.168.0.1
Port	Port of the database to connect	3306
Database Name	Name of the database to connect	dbname
Username	Username used for accessing the database. This user must have the permissions to read and write data tables and metadata.	cdm
Password	Password of the user	-
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-
Agent	The agent function will be unavailable soon and does not need to be configured.	-
Reference Sign	(Optional) Delimiter between the names of the referenced tables or columns. For details, see the product documentation of the corresponding database.	'
Driver Version	Select a driver version that adapts to the database type.	-
Fetch Size	(Optional) Displayed when you click **Show Advanced Attributes**. Number of rows obtained by each request. Set this parameter based on the data source and the job's data size. If the value is either too large or too small, the job may run for a long time.	1000

Parameter	Description	Example Value
Link Attributes	(Optional) Click **Add** to add the JDBC connector attributes of multiple specified data sources. For details, see the JDBC connector document of the corresponding database. The following are some examples: • **connectTimeout=360000** and **socketTimeout=360000**: When a large amount of data needs to be migrated or the entire table is retrieved using query statements, the migration fails due to connection timeout. In this case, you can customize the connection timeout interval (ms) and socket timeout interval (ms) to prevent failures caused by timeout.	sslmode=require

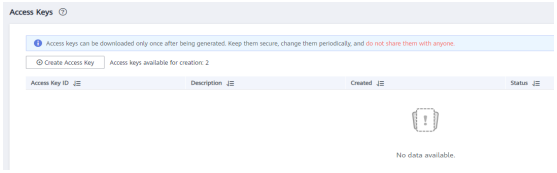
4.5.30 Link to CloudTable OpenTSDB

When connecting CDM to CloudTable OpenTSDB, configure the parameters as described in **Table 4-53**.

Table 4-53 CloudTable OpenTSDB link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	TSDB_link
OpenTSDB Link	ZK link of OpenTSDB	opentsdb-sp8afz7bgbps5ur.cloudtable.com:4242
Security Mode	Security or non-security mode If you select **Security**, enter the project ID, username, and AK/SK.	Nonsecurity

Parameter	Description	Example Value
Project ID	Project ID in the region where CloudTable resides A project is a group of tenant resources, and an account ID corresponds to the current account. The IAM ID corresponds to the current user. You can view the project IDs, account IDs, and user IDs in different regions on the corresponding pages. 1. Register with and log in to the management console. 2. Hover the cursor on the username in the upper right corner and select **My Credentials** from the drop-down list. 3. On the **API Credentials** page, obtain the account name, account ID, IAM username, and IAM user ID, and obtain the project ID from the project list.	-
Username	Username for accessing CloudTable	admin

Parameter	Description	Example Value
AK	AK and SK for accessing CloudTable. You need to create an access key for the current account and obtain an AK/SK pair. 1. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list. 2. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 4-30**. **Figure 4-30** Clicking Create Access Key  3. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**. **NOTE** • Only two access keys can be added for each user. • To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.	-
SK		-

4.6 Managing Jobs

4.6.1 Table/File Migration Jobs

Scenario

CDM supports table and file migration between homogeneous or heterogeneous data sources. For details about supported data sources, see **Supported Data Sources**.

Constraints

- The dirty data recording function depends on OBS.
- The JSON file of a job to be imported cannot exceed 1 MB.

- The size of a file to be transferred cannot exceed 1 TB.
- Field names of the source and destination parameters cannot contain ampersands (&) or number signs (%).

Prerequisites

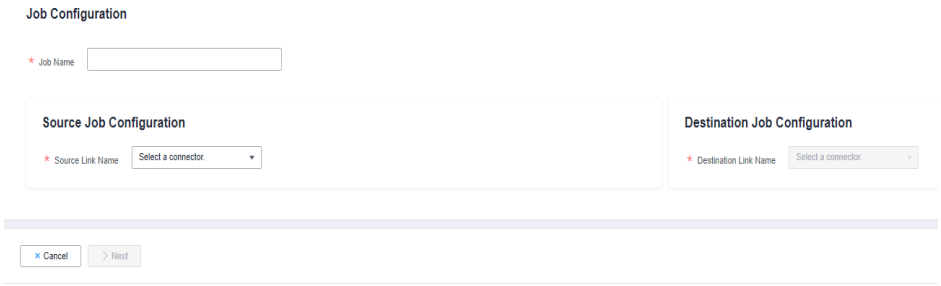
- A link has been created. For details, see **Creating a Link**.
- The CDM cluster can communicate with the data source.

Procedure

Step 1 Log in to the management console and choose **Service List** > **Cloud Data Migration**. In the left navigation pane, choose **Cluster Management**. Locate the target cluster and click **Job Management**.

Step 2 Choose **Table/File Migration** > **Create Job**. The page for configuring the job is displayed.

Figure 4-31 Creating a migration job

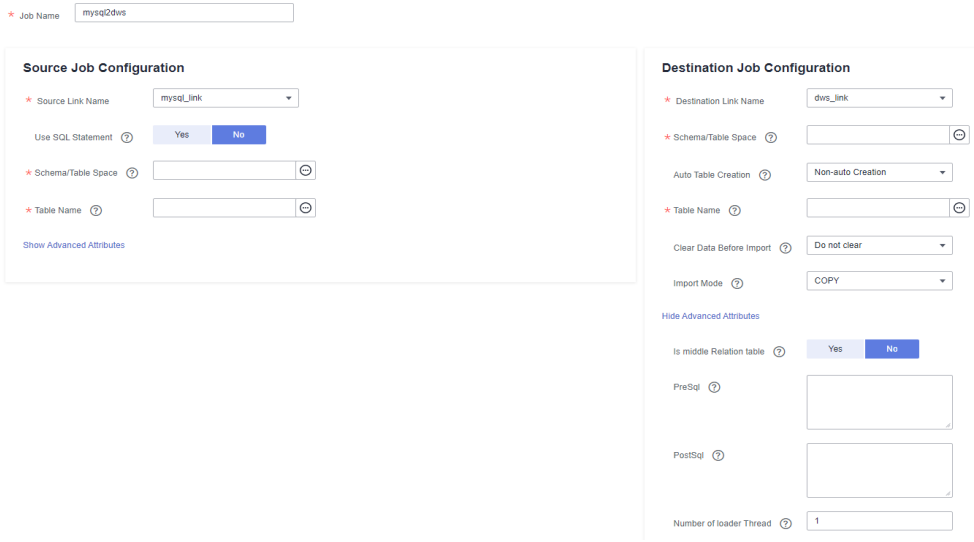


Step 3 Select the source and destination links.

- **Job Name**: Enter a string consisting of 1 to 240 characters. The name can contain digits, letters, hyphens (-), underscores (_), and periods (.), and cannot start with a hyphen (-) or period (.). An example value is **oracle2rds_t**.
- **Source Link Name**: Select the data source from which data will be exported.
- **Destination Link Name**: Select the data source to which data will be imported.

Step 4 Configure the source link parameters. **Figure 4-32** shows the job configurations for migrating MySQL to DWS.

Figure 4-32 Creating a job



The parameters vary with data sources. For details about the job parameters of other types of data sources, see **Table 4-54** and **Table 4-55**.

Table 4-54 Source link parameter description

Migration Source	Description	Parameter Settings
OBS	Data can be extracted in CSV, JSON, or binary format. Data extracted in binary format is free from file resolution, which ensures high performance and is more suitable for file migration.	For details, see **From OBS**.
• MRS HDFS • FusionInsight HDFS • Apache HDFS	HDFS data can be exported in CSV, Parquet, or binary format and can be compressed in multiple formats.	For details, see **From HDFS**.
• MRS HBase • FusionInsight HBase • Apache HBase • CloudTable Service	Data can be exported from MRS, FusionInsight HD, open source Apache Hadoop HBase, or CloudTable. You need to know all column families and field names of HBase tables.	For details, see **From HBase/ CloudTable**.

Migration Source	Description	Parameter Settings
• MRS Hive • FusionInsight Hive • Apache Hive	Data can be exported from Hive through the JDBC API. If the data source is Hive, CDM will automatically partition data using the Hive data partitioning file.	For details, see **From Hive**.
DLI	Data can be exported from DLI.	For details, see **From DLI**.
• FTP • SFTP	FTP and SFTP data can be exported in CSV, JSON, or binary format.	For details, see **From FTP/SFTP**.
• HTTP	These connectors are used to read files with an HTTP/HTTPS URL, such as reading public files on the third-party object storage system and web disks. Currently, data can only be exported from the HTTP URLs.	For details, see **From HTTP**.
Data Warehouse Service	Data can be exported from DWS.	For details, see **From DWS**.
SAP HANA	Data can be exported from SAP HANA.	For details, see **From SAP HANA**.
• RDS for PostgreSQL • RDS for SQL Server • Microsoft SQL Server • PostgreSQL	Data can be exported from the cloud database services. The non-cloud databases can be those created in the on-premises data center or deployed on ECSs, or database services on the third-party clouds.	When data is exported from these data sources, CDM uses the JDBC API to extract data. The job parameters for the migration source are the same. For details, see **From PostgreSQL/SQL Server**.
MySQL	Data can be exported from a MySQL database.	For details, see **From MySQL**.
Oracle	Data can be exported from an Oracle database.	For details, see **From Oracle**.

Migration Source	Description	Parameter Settings
Database Sharding	Data can be exported from a shard.	For details, see **From a Database Shard**.
• MongoDB • Document Database Service	Data can be exported from MongoDB or DDS.	For details, see **From MongoDB/DDS**.
Redis	Data can be exported from open source Redis.	For details, see **From Redis**.
Data Ingestion Service	Data can only be exported to Cloud Search Service (CSS).	For details, see **From DIS**.
• Apache Kafka • DMS Kafka • MRS Kafka	Data can only be exported to Cloud Search Service (CSS).	For details, see **From Kafka/DMS Kafka**.
• Cloud Search Service • Elasticsearch	Data can be exported from CSS or Elasticsearch.	For details, see **From Elasticsearch or CSS**.
MRS Hudi	Data can be exported from MRS Hudi.	For details, see **From MRS Hudi**.
MRS ClickHouse	Data can be exported from MRS ClickHouse.	For details, see **From MRS ClickHouse**.
ShenTong database	Data can be exported from a ShenTong database.	For details, see **From a ShenTong Database**.
Dameng database	Data can be exported from a Dameng database.	For details, see **From a Dameng Database**.

Step 5 Configure job parameters for the migration destination based on **Table 4-55**.

Table 4-55 Parameter description

Migration Destination	Description	Parameter Settings
OBS	Files (even in a large volume) can be batch migrated to OBS in CSV or binary format.	For details, see **To OBS**.

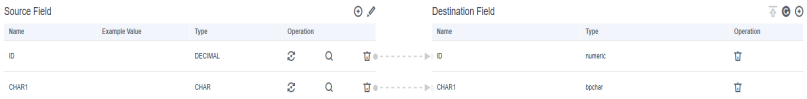
Migration Destination	Description	Parameter Settings
MRS HDFS	You can select a compression format when importing data to HDFS.	For details, see **To HDFS**.
MRS HBase CloudTable Service	Data can be imported to HBase. The compression algorithm can be set when a new HBase table is created.	For details, see **To HBase/ CloudTable**.
MRS Hive	Data can be rapidly imported to MRS Hive.	For details, see **To Hive**.
• MySQL • SQL Server • PostgreSQL	Data can be imported to cloud database services.	For details about how to use the JDBC API to import data, see **To MySQL/SQL Server/ PostgreSQL**.
DWS	Data can be imported to DWS.	For details, see **To DWS**.
Oracle	Data can be imported to an Oracle database.	For details, see **To Oracle**.
DLI	Data can be imported to DLI.	For details, see **To DLI**.
Elasticsearchor Cloud Search Service (CSS)	Data can be imported to CSS.	For details, see **To Elasticsearch/CSS**.
MRS Hudi	Data can be rapidly imported to MRS Hudi.	For details, see **To MRS Hudi**.
MRS ClickHouse	Data can be rapidly imported to MRS ClickHouse.	For details, see **To MRS ClickHouse**.
MongoDB	Data can be rapidly imported to MongoDB.	For details, see **To MongoDB**.

Step 6 After the parameters are configured, click **Next**. The **Map Field** tab page is displayed.

If files are migrated between FTP, SFTP, OBS, and HDFS and the migration source's **File Format** is set to **Binary**, files will be directly transferred, free from field mapping.

In other scenarios, CDM automatically maps fields of the source table and the destination table. You need to check whether the mapping and time format are correct. For example, check whether the source field type can be converted into the destination field type.

Figure 4-33 Field mapping

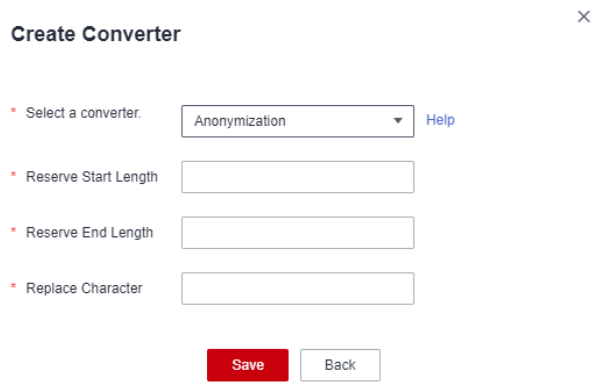


NOTE

- If the fields from the source and destination do not match, you can drag the fields to make adjustments.
- On the **Map Field** tab page, if CDM fails to obtain all columns by obtaining sample values (for example, when data is exported from HBase, CloudTable, or MongoDB, or when data is migrated from SFTP/FTP to DLI, there is a high probability that CDM failed to obtain all columns), you can click and select **Add a new field** to add new fields to ensure that the data imported to the migration destination is complete.
- When a relational database, Hive, DLI, or MRS Hudi is used as the migration source, sample values cannot be obtained.
- On the **Map Field** page, you can click to add custom constants, variables, and expressions.
- Column names are displayed when the source of the migration job is OBS, CSV files are to be migrated, and parameter **Extract first row as columns** is set to **Yes**.
- When SQLServer is the destination, fields of the timestamp type cannot be written. You must change their type (for example, to datatime) so that they can be written.
- When Hive serves as the source, data of the array and map types can be read.
- Field mapping is not involved when the binary format is used to migrate files to files.
- If the data is imported to DWS, you need to select the distribution columns in the destination fields. You are advised to select the distribution columns according to the following principles:
  1. Use the primary key as the distribution column.
  2. If multiple data segments are combined as primary keys, specify all primary keys as the distribution column.
  3. In the scenario where no primary key is available, if no distribution column is selected, DWS uses the first column as the distribution column by default. As a result, data skew risks exist.

Step 7 CDM supports field conversion. Click and then click **Create Converter**.

Figure 4-34 Creating a converter



CDM supports the following converters:

- **Anonymization**: hides key data in the character string.

  For example, if you want to convert **12345678910** to **123****8910**, configure the parameters as follows:

  - Set **Reserve Start Length** to **3**.
  - Set **Reserve End Length** to **4**.
  - Set **Replace Character** to *****.
- **Trim** automatically deletes the spaces before and after the character string.
- **Reverse string** automatically reverses a character string. For example, reverse **ABC** into **CBA**.
- **Replace string** replaces the specified character string.
- **Expression conversion** uses the JSP expression language (EL) to convert the current field or a row of data. For details, see **Field Conversion**.
- **Remove line break** deletes the newline characters, such as \n, \r, and \r\n from the field.

NOTE

If **Use SQL Statement** is set to **Yes** in the source job configuration, converters cannot be created.

Step 8 Click **Next**, set job parameters, and click **Show Advanced Attributes** to display and configure optional parameters.

Figure 4-35 Task parameters

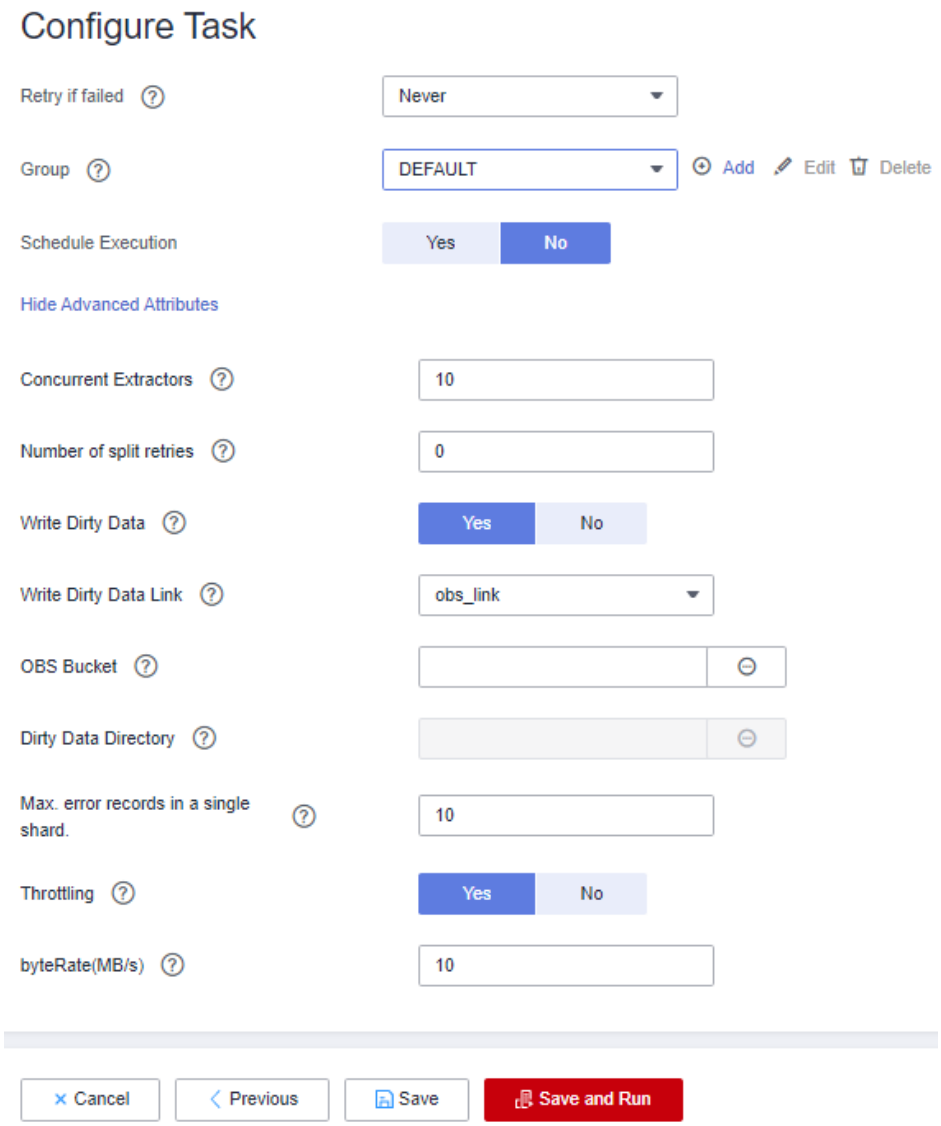


Table 4-56 describes related parameters.

Table 4-56 Parameter description

Parameter	Description	Example Value
Retry upon Failure	You can select **Retry 3 times** or **Never**. You are advised to configure automatic retry for only file migration jobs or database migration jobs with **Import to Staging Table** enabled to avoid data inconsistency caused by repeated data writes. **NOTE** If you want to set parameters in DataArts Studio DataArts Factory to schedule the CDM migration job, do not configure this parameter. Instead, set parameter **Retry upon Failure** for the CDM node in DataArts Factory.	Never
Job	Select a group where the job resides. The default group is **DEFAULT**. On the **Job Management** page, jobs can be displayed, started, or exported by group.	DEFAULT
Schedule Execution	If you select **Yes**, you can set the start time, cycle, and validity period of a job. For details, see **Scheduling Job Execution**. **NOTE** If you use DataArts Studio DataArts Factory to schedule the CDM migration job and configure this parameter, both configurations take effect. To ensure unified service logic and avoid scheduling conflicts, enable job scheduling in DataArts Factory and do not configure a scheduled task for the job in DataArts Migration.	No

Parameter	Description	Example Value
Concurrent Extractors	Configure the number of tasks to be split from a CDM job. CDM migrates data through data migration jobs. It works in the following way: 1. When data migration jobs are submitted, CDM splits each job into multiple tasks based on the **Concurrent Extractors** parameter in the job configuration. **NOTE** Jobs for different data sources may be split based on different dimensions. Some jobs may not be split based on the **Concurrent Extractors** parameter. 2. CDM submits the tasks to the running pool in sequence. Tasks (defined by **Maximum Concurrent Extractors**) run concurrently. Excess tasks are queued. By setting appropriate values for this parameter and the **Maximum Concurrent Extractors** parameter, you can accelerate migration. Configure the number of concurrent extractors based on the following rules: 1. When data is to be migrated to files, CDM does not support multiple concurrent tasks. In this case, set a single process to extract data. 2. If each row of the table contains less than or equal to 1 MB data, data can be extracted concurrently. If each row contains more than 1 MB data, it is recommended	1

Parameter	Description	Example Value
	that data be extracted in a single thread. 3. Set **Concurrent Extractors** for a job based on **Maximum Concurrent Extractors** for the cluster. It is recommended that **Concurrent Extractors** is less than **Maximum Concurrent Extractors**. 4. If the destination is DLI, you are advised to set the number of concurrent extractors to 1. Otherwise, data may fail to be written. The maximum number of concurrent extractors for a cluster varies depending on the CDM cluster flavor. You are advised to set the maximum number of concurrent extractors to twice the number of vCPUs of the CDM cluster. For example, the maximum number of concurrent extractors for a cluster with 8 vCPUs and 16 GB memory is 16.	
Concurrent Loaders	Number of Loaders to be concurrently executed This parameter is displayed only when HBase or Hive serves as the destination data source.	3
Number of split retries	Number of retries when a split fails to be executed. Value **0** indicates that no retry will be performed.	0

Parameter	Description	Example Value
Write Dirty Data	Whether to record dirty data. By default, this parameter is set to **No**. Dirty data in CDM refers to the data in invalid format. If the source data contains dirty data, you are advised to enable this function. Otherwise, the migration job may fail. **NOTE** Dirty data can only be written to OBS paths. Therefore, this parameter is available only when an OBS link is available.	Yes
Write Dirty Data Link	This parameter is displayed only when **Write Dirty Data** is set to **Yes**. You can only select an OBS link.	obs_link
OBS Bucket	This parameter is displayed only when **Write Dirty Data Link** is a link to OBS. Name of the OBS bucket to which the dirty data will be written.	dirtydata
Dirty Data Directory	This parameter is displayed only when **Write Dirty Data** is set to **Yes**. Dirty data is stored in the directory for storing dirty data on OBS. Dirty data is saved only when this parameter is configured. You can go to this directory to query data that fails to be processed or is filtered out during job execution, and check the source data that does not meet conversion or cleaning rules.	/user/dirtydir

Parameter	Description	Example Value
Max. Error Records in a Single Shard	This parameter is displayed only when **Write Dirty Data** is set to **Yes**. When the number of error records of a single map exceeds the upper limit, the job will automatically terminate and the imported data cannot be rolled back. You are advised to use a temporary table as the destination table. After the data is imported, rename the table or combine it into the final data table.	0
Throttling	Enabling throttling reduces the read pressure on the source. It controls the CDM transmission rate, not the NIC traffic. **NOTE** • Throttling can be enabled for non-binary file migration jobs. • To configure throttling for multiple jobs, multiply the rate by the number of concurrent jobs. • Throttling is not supported for binary transmission between files.	Yes
Max. error records in a single shard	Maximum rate for a job. To configure throttling for multiple jobs, multiply the rate by the number of concurrent jobs. **NOTE** The rate is an integer greater than 1.	20

Parameter	Description	Example Value
Intermediate Queue Cache Size (MB)	Amount of data that the intermediate queue can cache. The value ranges from 1 to 500. The default value is **64**. If the amount of data of a row exceeds the value of this parameter, the migration may fail. If the value of this parameter is too large, the cluster may not run properly. Set an appropriate value for this parameter and use the default value (**64**) unless otherwise specified.	64

Step 9 Click **Save** or **Save and Run**. On the displayed page, you can view the job status.

NOTE

The job status can be **New**, **Pending**, **Booting**, **Running**, **Failed**, **Succeeded**, or **Stopped**.

Pending indicates that the job is waiting to be scheduled by the system, and **Booting** indicates that the data to be migrated is being analyzed.

----End

4.6.2 Creating an Entire Database Migration Job

Scenario

CDM supports entire DB migration between homogeneous and heterogeneous data sources. The migration principles are the same as those in **Table/File Migration Jobs**. Each type of Elasticsearch, each key prefix of Redis, or each collection of MongoDB can be executed concurrently as a subtask.

NOTE

Each time an entire DB migration job is executed, its subtasks are recreated based on the configuration of the migration job. You cannot modify the subtasks and then run the migration job again.

Supported Data Sources lists the data sources supporting entire database migration.

Constraints

Field names of the source and destination parameters cannot contain ampersands (&) or number signs (%).

Prerequisites

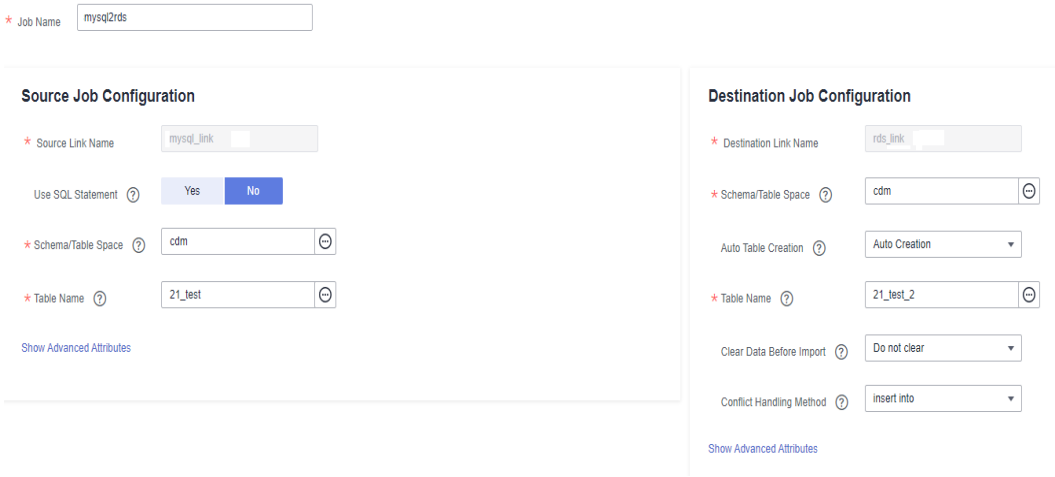
- A link has been created. For details, see **Creating a Link**.
- The CDM cluster can communicate with the data source.

Procedure

Step 1 Log in to the management console and choose **Service List** > **Cloud Data Migration**. In the left navigation pane, choose **Cluster Management**. Locate the target cluster and click **Job Management**.

Step 2 Choose **Entire DB Migration** > **Create Job**. The page for configuring the job is displayed.

Figure 4-36 Creating an entire DB migration job



Step 3 Configure the related parameters of the source database according to **Table 4-57**.

Table 4-57 Parameter description

Source Database	Parameter	Description	Example Value
• DWS • MySQL • PostgreSQL • SQL Server • Oracle • SAP HANA	Schema/ Tablespace	Name of the schema or tablespace from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting a schema or directly enter a schema or tablespace. If the desired schema or tablespace is not displayed, check whether the login account has the permissions required to query metadata.	schema
	WHERE Clause	WHERE clause used to specify the tables to be extracted. This parameter applies to all subtables in the entire DB migration. If this parameter is not set, the entire table is extracted. If the table to be migrated does not contain the fields specified by the WHERE clause, the migration will fail. You can set a date macro variable to extract data generated on a specific date. For details, see **Incremental Migration of Relational Databases**.	age > 18 and age <= 60
	Null in Partition Column	Whether a partition field can be null	Yes

Source Database	Parameter	Description	Example Value
Hive	Database Name	Name of the database to be migrated. The user configured in the source link must have the permission to read the database.	hivedb
HBase CloudTable	Start Time	Start time (included). The format is *yyyy-MM-dd hh:mm:ss*. The dateformat time macro variable function is supported. Examples: **2017-12-31 20:00:00**, **${dateformat(yyyy-MM-dd, -1, DAY)} 02:00:00**, and **${dateformat(yyyy-MM-dd HH:mm:ss, -1, DAY)}**	"2017-12-31 20:00:00"
	End Time	End time (excluded) The format is *yyyy-MM-dd hh:mm:ss*. The dateformat time macro variable function is supported. Examples: **2018-01-01 20:00:00**, **${dateformat(yyyy-MM-dd, -1, DAY)} 02:00:00**, and **${dateformat(yyyy-MM-dd HH:mm:ss, -1, DAY)}**	"2018-01-01 20:00:00"
Redis	Key Filter Character	Filter character used to determine the keys to be migrated For example, if the value of this parameter is **a***, all asterisks (*) will be migrated.	a*

Source Database	Parameter	Description	Example Value
DDS	Database Name	Name of the database from which data is to be migrated. The user configured in the source link must have the permission to read the database.	ddsdb
	Query Filter	Filter used to match documents. Example: **{HTTPStatusCode:{$gt:"400",$lt:"500"},HTTPMethod:"GET"}**	-

Step 4 Configure the related parameters, from **Table 4-58**, for the destination cloud service.

Table 4-58 Destination job parameters

Destination Database	Parameter	Description	Example Value
• RDS for MySQL • RDS for PostgreSQL • RDS for SQL Server	-	For details about the destination job parameters required for entire DB migration to an RDS database, see **To MySQL/SQL Server/PostgreSQL**.	schema
DWS	-	For details about the destination job parameters required for entire DB migration to DWS, see **To DWS**.	-
MRS Hive	-	For details about the destination job parameters required for entire DB migration to MRS HIVE, see **To Hive**.	hivedb
MRS HBase CloudTable	-	For details about the destination job parameters required for entire DB migration to MRS HBase or CloudTable, see **To HBase/CloudTable**.	Yes
Redis	Clear Database	Clears the database data before data import.	Yes

Destination Database	Parameter	Description	Example Value
DDS	Database Name	Name of the database from which data is to be migrated. The user configured in the source link must have the permission to read the database.	mongodb
	Migration Behavior	Select **Add** or **Replace**.	-

Step 5 If a relational database is migrated, after job parameters are configured, click **Next** to access the page for selecting tables. You can select the tables to be migrated to the migration destination based on your requirements.

Step 6 Click **Next** and set job parameters.

Figure 4-37 Task parameters

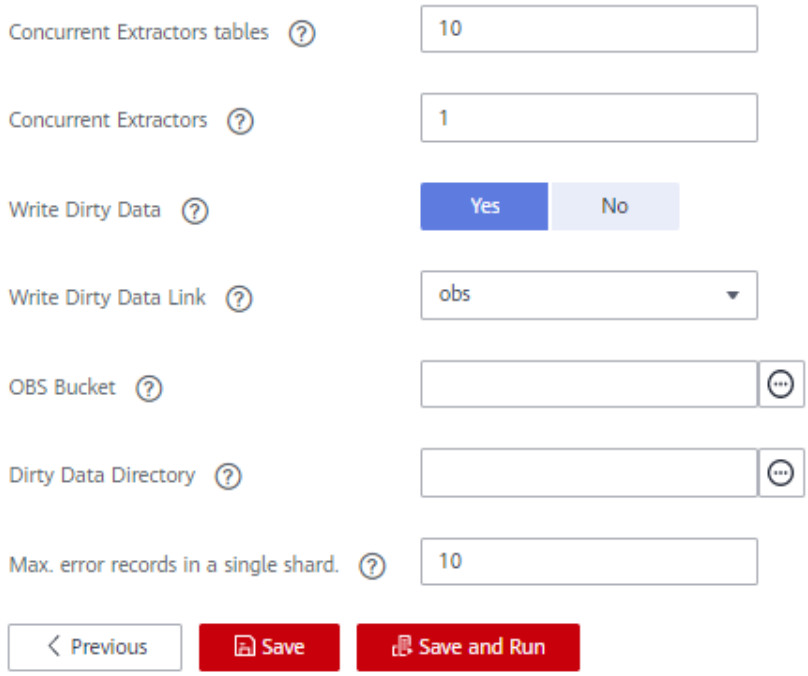


Table 4-59 describes related parameters.

Table 4-59 Task configuration parameters

Parameter	Description	Example Value
Concurrent Tables	Number of tables to be concurrently executed	3

Parameter	Description	Example Value
Concurrent Extractors	Number of extractors to be concurrently executed. Generally, retain the default value.	1
Write Dirty Data	Whether to record dirty data. By default, this parameter is set to **No**.	Yes
Write Dirty Data Link	This parameter is only displayed when **Write Dirty Data** is set to **Yes**. Only links to OBS support dirty data writes.	obs_link
OBS Bucket	This parameter is only displayed when **Write Dirty Data Link** is a link to OBS. Name of the OBS bucket to which the dirty data will be written.	dirtydata
Dirty Data Directory	This parameter is only displayed when **Write Dirty Data** is set to **Yes**. Directory for storing dirty data on OBS. Dirty data is saved only when this parameter is configured. You can go to this directory to query data that fails to be processed or is filtered out during job execution, and check the source data that does not meet conversion or cleaning rules.	/user/dirtydir
Max. Error Records in a Single Shard	This parameter is only displayed when **Write Dirty Data** is set to **Yes**. When the number of error records of a single map exceeds the upper limit, the job will automatically terminate and the imported data cannot be rolled back. You are advised to use a temporary table as the destination table. After the data is imported, rename the table or combine it into the final data table.	0

Step 7 Click **Save** or **Save and Run**.

When the job starts running, a sub-job will be generated for each table. You can click the job name to view the sub-job list.

----End

NOTE

During the migration of an entire Oracle database to Hudi, if you select a view or a table that has no primary key at the source, automatic table creation is not supported.

4.6.3 Source Job Parameters

4.6.3.1 From OBS

If the source link of a job is an **OBS link**, configure the source job parameters based on **Table 4-60**.

Advanced attributes are optional and not displayed by default. You can click **Show Advanced Attributes** to display them.

Table 4-60 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	Bucket Name	Name of the bucket from which data will be migrated	BUCKET_2
	Source Directory/File	This parameter is available only when **Pull List File** is set to **No**. Directory or file path from which data will be extracted. You can enter a maximum of 50 file paths. By default, the file paths are separated by vertical bars (|). You can also customize a file separator. For details, see **Migration of a List of Files**. Directory from which data is to be migrated. All files (including all nested subdirectories and their subfiles) in the directory will be migrated. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	FROM/example.csv

Category	Parameter	Description	Example Value
	File Format	Format in which CDM parses data. The options are as follows: • **CSV**: Source files will be migrated to tables after being converted to CSV format. • **Binary**: Files (even not in binary format) will be transferred directly. It is used for file copy. • **JSON**: Source files will be migrated to tables after being converted to JSON format.	CSV
	Pull List File	This parameter is displayed only when **File Format** is set to **Binary**. If the pull list file function is enabled, the content of a file (such as a .txt file) in an OBS bucket can be read as the list of files to be migrated. The content in the file must be the absolute path of the file to be migrated (rather than a directory). For example, the content is as follows: /052101/DAY20211110.data /052101/DAY20211111.data	Yes
	OBS Link of List File	This parameter is available only when **Pull List File** is set to **Yes**. You can select the OBS link where the list file is located.	OBS_test_link
	OBS Bucket of entries files	This parameter is available only when **Pull List File** is set to **Yes**. It indicates the name of the OBS bucket where the list file is located.	01
	Path/ Directory of entries files	This parameter is available only when **Pull List File** is set to **Yes**. It indicates the absolute path or directory of the list file in the OBS bucket. You are advised to select the absolute path of the file. If you select a directory, files in subdirectories can also be migrated. However, if the number of files in the directory is too large, the cluster memory may become insufficient.	/0521/ Lists.txt

Category	Parameter	Description	Example Value
	JSON Type	This parameter is displayed only when **File Format** is set to **JSON**. Type of a JSON object stored in a JSON file. The options are **JSON object** and **JSON array**.	JSON object
	JSON Reference Node	This parameter is used only when **File Format** is set to **JSON** and **JSON Type** is set to **JSON Object**. CDM parses the data under the JSON node. If the node's corresponding data is a JSON array, the system will extract data from the array in the same pattern. Use periods (.) to separate multi-layer nested JSON nodes.	data.list
Advanced attributes	Line Separator	Lind feed character in a file. By default, the system automatically identifies **\n**, **\r**, and **\r\n**. This parameter is displayed only when **File Format** is set to **CSV**.	\n
	Field Delimiter	Character used to separate fields in the file. To set the **Tab** key as the delimiter, set this parameter to **\t**. This parameter is displayed only when **File Format** is set to **CSV**.	,
	Use Quote Character	If you set this parameter to **Yes**, the field delimiters in the encircling symbol are regarded as a part of the string value. Currently, the default encircling symbol of CDM is **"**.	No
	Using Escape Char	If you select **Yes**, the backslash (\) in the data row is used as an escape character. If you select **No**, the backslash (\) in the CSV file will not be escaped. CSV supports only the backslash (\) as the escape character.	Yes
	Use RE to Separate Fields	Whether to use regular expressions to separate fields. If you set this parameter to **Yes**, **Field Delimiter** becomes invalid. This parameter is displayed only when **File Format** is set to **CSV**.	Yes
	Regular Expression	Regular expression used to separate fields. For details about regular expressions, see **Regular Expressions for Separating Semi-structured Text**.	^(\d.*\d) (\w*) \[(.*) \] ([\w\.]*) (\w.*).*

Category	Parameter	Description	Example Value
	Use First N Rows as Header	This parameter is displayed only when **File Format** is set to **CSV**. When you migrate a CSV file to a table, CDM writes all data to the table by default. If you set this parameter to **Yes**, CDM uses the first N rows of the CSV file as the heading row and does not write the row to the destination table.	No
	The Number of Header Rows	This parameter is available when **Use First N Rows as Header** is set to **Yes**. It specifies the number of header rows to be skipped during data extraction. **NOTE** The number of header rows cannot be empty. The value is an integer from 1 to 99.	1
	Extract first row as columns	This parameter is available when **Use First N Rows as Header** is set to **Yes**. It specifies whether to parse the first row of the header as a column name. The column name is displayed in the source field during field mapping configuration. **NOTE** • If the number of header rows is greater than 1, only the first row of the header can be parsed as the column name. • The column name cannot contain the ampersand (&). Otherwise, the job migration fails. If the column name contains the ampersand (&), you must change it in the CSV file to ensure successful migration.	Yes
	Encoding Type	Encoding type, for example, **UTF-8** or **GBK**. You can set the encoding type for text files only. This parameter is invalid when **File Format** is set to **Binary**.	GBK

Category	Parameter	Description	Example Value
	Compression Format	The options are as follows: • **NONE**: Files in all formats can be transferred. • **GZIP**: Only files in gzip format can be transferred. • **ZIP**: Only files in Zip format can be transferred. • **TAR.GZ**: Files in TAR.GZ format are transferred.	NONE
	Compressed File Suffix	This parameter is displayed when **Compression Format** is not **NONE**. This parameter specifies the extension of the files to be decompressed. The decompression operation is performed only when the file name extension is used in a batch of files. Otherwise, files are transferred in the original format. If you enter ***** or leave the parameter blank, all files are decompressed.	*
	Start Job by Marker File	Whether to start a job by a marker file. A job is only started if there is a marker file for starting the job in the source path. If there is no marker file, the job will be suspended for a period of time specified by **Suspension Period**.	No
	Marker File	Name of the marker file for starting a job. If you specify a marker file, the migration job is executed only when the marker file exists in the source path. The marker file will not be migrated.	ok.txt
	Suspension Period	Waiting period for a marker file. If you set **Start Job by Marker File** to **Yes** but there is no marker file in the source path, the job fails when the suspension period times out. If you set this parameter to **0** and there is no marker file in the source path, the job will fail immediately. Unit: second	10

Category	Parameter	Description	Example Value
	File Separator	File separator. If you enter multiple file paths in **Source Directory/Files**, CDM uses the file separator to identify files. The default value is **|**.	|
	Filter Type	Only paths or files that meet the filtering conditions are transferred. The options are **None**, **Wildcard**, and **Regex**. For details, see **Incremental File Migration**.	Wildcard
	Directory Filter	If you set **Filter Type** to **Wildcard** or **Regex**, enter a wildcard character to filter paths. The paths that meet the filtering condition are migrated. You can configure multiple paths separated by commas (,). **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	*input
	File Filter	If you set **Filter Type** to **Wildcard** or **Regex**, you can enter a wildcard character to search for files in a specified path. The files that meet the search criteria are migrated. You can configure multiple files separated by commas (,). **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	*.csv,*.txt
	Time Filter	If you select **Yes**, files are transferred based on their modification time.	Yes

Category	Parameter	Description	Example Value
	Minimum Timestamp	If you set **Filter Type** to **Time Filter**, and specify a point in time for this parameter, only the files modified at or after the specified time are transferred. The time format must be *yyyy-MM-dd HH:mm:ss*. This parameter can be set to a macro variable of date and time. For example, **${timestamp(dateformat(yyyy-MM-dd HH:mm:ss,-90,DAY))}** indicates that only files generated within the latest 90 days are migrated. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	2019-06-01 00:00:00
	Maximum Timestamp	If you set **Filter Type** to **Time Filter**, and specify a point in time for this parameter, only the files modified before the specified time are transferred. The time format must be *yyyy-MM-dd HH:mm:ss*. This parameter can be set to a macro variable of date and time. For example, **${timestamp(dateformat(yyyy-MM-dd HH:mm:ss))}** indicates that only the files whose modification time is earlier than the current time are migrated. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	2019-07-01 00:00:00
	Disregard Non-existent Path or File	If this is set to **Yes**, the job can be successfully executed even if the source path does not exist.	No

Category	Parameter	Description	Example Value
	MD5 File Extension	This parameter is displayed only when **File Format** is set to **Binary**. This parameter is used to check whether the files extracted by CDM are consistent with source files. For details, see **MD5 Verification**.	.md5

NOTE

1. CDM supports incremental file migration (by skipping repeated files), but does not support resumable transfer.

   For example, if three files are to be migrated and the second file fails to be migrated due to the network fault. When the migration task is started again, the first file is skipped. The second file, however, cannot be migrated from the point where the fault occurs, but can only be migrated again.
2. During file migration, a single task supports millions of files. If there are too many files in the directory to be migrated, you are advised to split the files into different directories and create multiple tasks.

4.6.3.2 From HDFS

If the source link of a job is an **HDFS link**, that is, if data is exported from MRS HDFS, FusionInsight HDFS, or Apache HDFS, configure the source job parameters based on **Table 4-61**.

Table 4-61 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	Source Link Name	Select a type from the drop-down list box.	hdfs_to_cdm

Category	Parameter	Description	Example Value
	Source Directory/ File	This parameter is available only when **Pull List File** is set to **No**. Directory or file path from which data will be extracted. Directory from which data is to be migrated. All files (including all nested subdirectories and their subfiles) in the directory will be migrated. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	/user/cdm/
	File Format	File format used when transferring data. The options are as follows: • **CSV**: Source files will be migrated to tables after being converted to CSV format. • **Binary**: Files (even not in binary format) will be transferred directly. It is used for file copy. • **Parquet**: Source files will be migrated to tables after being converted to Parquet format.	CSV

Category	Parameter	Description	Example Value
	Pull List File	This parameter is displayed only when **File Format** is set to **Binary**. If the pull list file function is enabled, the content of a file (such as a .txt file) in an OBS bucket can be read as the list of files to be migrated. The content in the file must be the absolute path of the file to be migrated (rather than a directory). The following is example content: /mrs/job-properties/application_1634891604621_0014/job.properties /mrs/job-properties/application_1634891604621_0029/job.properties	Yes
	OBS Link of List File	This parameter is available only when **Pull List File** is set to **Yes**. You can select the OBS link where the list file is located.	OBS_test_link
	OBS Bucket of entries files	This parameter is available only when **Pull List File** is set to **Yes**. It indicates the name of the OBS bucket where the list file is located.	01
	Path/Directory of entries files	This parameter is available only when **Pull List File** is set to **Yes**. It indicates the absolute path or directory of the list file in the OBS bucket.	/0521/Lists.txt
Advanced attributes	Line Separator	Lind feed character in a file. By default, the system automatically identifies **\n**, **\r**, and **\r\n**. This parameter is displayed only when **File Format** is set to **CSV**.	\n
	Field Delimiter	Character used to separate fields in the file. To set the **Tab** key as the delimiter, set this parameter to **\t**. This parameter is displayed only when **File Format** is set to **CSV**.	,

Category	Parameter	Description	Example Value
	Use First Row as Header	This parameter is displayed only when **File Format** is set to **CSV**. When you migrate a CSV file to a table, CDM writes all data to the table by default. If you set this parameter to **Yes**, CDM uses the first N rows of the CSV file as the heading row and does not write the row to the destination table.	No
	Encoding Type	Encoding type, for example, **UTF-8** or **GBK**. You can set the encoding type for text files only. This parameter is invalid when **File Format** is set to **Binary**.	GBK
	Start Job by Marker File	Whether to start a job by a marker file. A job is only started if there is a marker file for starting the job in the source path. If there is no marker file, the job will be suspended for a period of time specified by **Suspension Period**.	ok.txt
	Filter Type	Only paths or files that meet the filtering conditions are transferred. The options are **None**, **Wildcard**, and **Regex**. For details, see **Incremental File Migration**.	-

Category	Parameter	Description	Example Value
	Directory Filter	If you set **Filter Type** to **Wildcard** or **Regex**, enter a wildcard character to filter paths. The paths that meet the filtering condition are migrated. You can configure multiple paths separated by commas (,). **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	*input
	File Filter	If you set **Filter Type** to **Wildcard** or **Regex**, you can enter a wildcard character to search for files in a specified path. The files that meet the search criteria are migrated. You can configure multiple files separated by commas (,). **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	*.csv
	Time Filter	If you select **Yes**, files are transferred based on their modification time.	Yes

Category	Parameter	Description	Example Value
	Minimum Timestamp	If you set **Filter Type** to **Time Filter**, and specify a point in time for this parameter, only the files modified at or after the specified time are transferred. The time format must be *yyyy-MM-dd HH:mm:ss*. This parameter can be set to a macro variable of date and time. For example, **${timestamp(dateformat(yyyy-MM-dd HH:mm:ss,-90,DAY))}** indicates that only files generated within the latest 90 days are migrated. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	2019-07-01 00:00:00

Category	Parameter	Description	Example Value
	Maximum Timestamp	If you set **Filter Type** to **Time Filter**, and specify a point in time for this parameter, only the files modified before the specified time are transferred. The time format must be *yyyy-MM-dd HH:mm:ss*. This parameter can be set to a macro variable of date and time. For example, **${timestamp(dateformat(yyyy-MM-dd HH:mm:ss))}** indicates that only the files whose modification time is earlier than the current time are migrated. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	2019-07-30 00:00:00
	Create Snapshot	If you set this parameter to **Yes**, CDM creates a snapshot for the source directory to be migrated (the snapshot cannot be created for a single file) before it reads files from HDFS. Then CDM migrates the data in the snapshot. Only the HDFS administrator can create a snapshot. After the CDM job is completed, the snapshot is deleted.	No

Category	Parameter	Description	Example Value
	Encryption	This parameter is displayed only when **File Format** is set to **Binary**. If the source data is encrypted, CDM can decrypt the data before exporting it. Select whether to decrypt the source data and select a decryption algorithm. The options are as follows: • **NONE**: Export data without decrypting it. • **AES-256-GCM**: The AES 256-bit encryption algorithm is used to encrypt data. Currently, only the AES-256-GCM (NoPadding) encryption algorithm is supported. This parameter is used for encryption at the migration destination and decryption at the migration source. For details, see **Encryption and Decryption During File Migration**.	AES-256-GCM
	DEK	This parameter is displayed only when **Encryption** is set to **AES-256-GCM**. The key consists of 64 hexadecimal numbers and must be the same as the **DEK** configured during encryption. If the encryption and decryption keys are inconsistent, the system does not report an exception, but the decrypted data is incorrect.	DD0AE00DFECD78BF051BCFDA25BD4E320DB0A7AC75A1F3FC3D3C56A457DCDC1B

Category	Parameter	Description	Example Value
	IV	This parameter is displayed only when **Encryption** is set to **AES-256-GCM**. The initialization vector consists of 32 hexadecimal numbers and must be the same as the **IV** configured during encryption. If the encryption and decryption keys are inconsistent, the system does not report an exception, but the decrypted data is incorrect.	5C91687BA886EDCD12ACBC3FF19A3C3F
	MD5 File Extension	This parameter is displayed only when **File Format** is set to **Binary**. This parameter is used to check whether the files extracted by CDM are consistent with source files. For details, see **MD5 Verification**.	.md5

4.6.3.3 From HBase/CloudTable

If the source link of a job is an **HBase** or **CloudTable** link, that is, if data is exported from MRS HBase, FusionInsight HBase, CloudTable, or Apache HBase, configure the source job parameters based on **Table 4-62**.

NOTE

1. When you migrate data from CloudTable or HBase, CDM reads the first row of the table as an example of the field list. If the first row of data does not contain all fields of the table, you need to manually add fields.
2. Because HBase is schema-less, CDM cannot obtain the data types. If the data is stored in binary format, CDM cannot parse the data.
3. When data is exported from HBase or CloudTable, because HBase/CloudTable is schema-less storage systems, CDM requires that the source numeric fields be stored in regular decimal format rather than in binary format. For example, the value 100 needs to be stored as **100** rather than **01100100**.

Table 4-62 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	Table Name	Name of the HBase table that data will be exported from This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	TBL_2
	Column Families	(Optional) Column families to which the exported data belongs	CF1&CF2
Advanced attributes	Split Rowkey	(Optional) Whether to split a rowkey. The default value is **No**.	Yes
	Rowkey Delimiter	(Optional) Delimiter used to split a rowkey. If this parameter is left empty, the rowkey will not be split.	|

Category	Parameter	Description	Example Value
	Start Time	(Optional) Start time (including the value) for extracting data. The format is *yyyy-MM-dd HH:mm:ss*. Only the data generated at the specified time and later is extracted. This parameter can be set to a macro variable of date and time. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	2019-01-01 20:00:00
	End Time	(Optional) End time (excluding the value) for extracting data. The format is *yyyy-MM-dd HH:mm:ss*. Only the data generated before the time point is extracted. This parameter can be set to a macro variable of date and time. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	2019-02-01 20:00:00

4.6.3.4 From Hive

If the source link of a job is a **Hive link**, configure the source job parameters based on **Table 4-63**.

Table 4-63 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	Database Name	Database name. Click the icon next to the text box. The dialog box for selecting the database is displayed.	default
	Table Name	Hive table name. Click the icon next to the text box. The dialog box for selecting the table is displayed. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	TBL_E

Category	Parameter	Description	Example Value
	Read Mode	Two read modes are available: HDFS and JDBC. By default, the HDFS mode is used. If you do not need to use the WHERE condition to filter data or add new fields on the field mapping page, select the HDFS mode. • The HDFS mode shows good performance, but in this mode, you cannot use the WHERE condition to filter data or add new fields on the field mapping page. • The HDFS mode allows you to use the WHERE condition to filter data or add new fields on the field mapping page.	HDFS
	Use SQL Statement	Whether you can use SQL statements to export data from a relational database	No

Category	Parameter	Description	Example Value

	SQL Statement	When **Use SQL Statement** is set to **Yes**, enter an SQL statement here. CDM exports data based on the SQL statement.
		NOTE
		● SQL statements can only be used to query data. Join and nesting are supported, but multiple query statements are not allowed, for example, **select * from table a; select * from table b**.
		● With statements are not supported.
		● Comments, such as **--** and **/***, are not supported.
		● Addition, deletion, and modification operations are not supported, including but not limited to the following:
		  ● load data
		  ● delete from
		  ● alter table
		  ● create table
		  ● drop table
		  ● into outfile
			select id,name from sqoop.user;

Category	Parameter	Description	Example Value
Advanced attributes	Partition Values	This parameter is displayed when you select the HDFS read mode and click **Show Advanced Attributes**. This parameter indicates extracting the partition of a specified value. The attribute name is the partition name. You can configure multiple values (separated by spaces) or a field value range. The time macro function is supported. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	• Attribute value in the single-value or multi-value filtering scenario: ${dateformat(yyyyMMdd, -1, DAY)} ${dateformat(yyyyMMdd)} • Attribute value in the range filtering scenario: ${value} >= ${dateformat(yyyyMMdd, -7, DAY)} && ${value} < ${dateformat(yyyyMMdd)}

Category	Parameter	Description	Example Value
	WHERE Clause	This parameter is displayed when you select the JDBC read mode and click **Show Advanced Attributes**. This parameter indicates the WHERE clause to be extracted. If this parameter is not set, the entire table is extracted. If the table to be migrated does not contain the fields specified by the WHERE clause, the migration will fail. You can set a date macro variable to extract data generated on a specific date. For details, see **Incremental Migration of Relational Databases**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	age > 18 and age <= 60

NOTE

If the data source is Hive, CDM will automatically partition data using the Hive data partitioning file.

4.6.3.5 From DLI

If the source link of a job is a **DLI link**, configure the source job parameters based on **Table 4-64**.

Table 4-64 Parameter description

Parameter	Description	Example Value
Resource Queue	Resource queue to which the destination table belongs The default queue of DLI cannot be used for migration jobs. You need to create a SQL queue in DLI.	cdm
Database Name	Name of the database to which data will be written	dli
Table Name	Name of the table to which data will be written	car_detail
Partition	Partition information	year=2020,location=sun

4.6.3.6 From FTP/SFTP

If the source link of a job is an **FTP or SFTP link**, configure the source job parameters based on **Table 4-65**.

Advanced attributes are optional and not displayed by default. You can click **Show Advanced Attributes** to display them.

Table 4-65 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	Source Directory/File	Directory or file path from which data will be extracted. You can enter a maximum of 50 file paths. By default, the file paths are separated by vertical bars (|). You can also customize a file separator. For details, see **Migration of a List of Files**. Directory from which data is to be migrated. All files (including all nested subdirectories and their subfiles) in the directory will be migrated. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	/ftp/a.csv|/ftp/b.txt
	File Format	Format in which CDM parses data. The options are as follows: • **CSV**: Source files will be migrated to tables after being converted to CSV format. • **Binary**: Files (even not in binary format) will be transferred directly. This format is used to copy data from a file to another. • **JSON**: Source files will be migrated to tables after being converted to JSON format. **NOTE** If the destination is OBS, only the binary format is supported.	CSV
	JSON Type	This parameter is displayed only when **File Format** is set to **JSON**. Type of a JSON object stored in a JSON file. The options are **JSON object** and **JSON array**.	JSON object

Category	Parameter	Description	Example Value
	JSON Reference Node	This parameter is used only when **File Format** is set to **JSON** and **JSON Type** is set to **JSON Object**. CDM parses the data under the JSON node. If the node's corresponding data is a JSON array, the system will extract data from the array in the same pattern. Use periods (.) to separate multi-layer nested JSON nodes.	data.list
Advanced attributes	Use rfc4180 Parser	This parameter is displayed only when **File Format** is set to **CSV**. It specifies whether to use the rfc4180 parser to parse CSV files.	No
	Line Separator	Lind feed character in a file. By default, the system automatically identifies **\n**, **\r**, and **\r\n**. This parameter is displayed only when **File Format** is set to **CSV**.	\n
	Field Delimiter	Character used to separate fields in the file. To set the **Tab** key as the delimiter, set this parameter to **\t**. This parameter is displayed only when **File Format** is set to **CSV**.	,
	Use Quote Character	If you set this parameter to **Yes**, the field delimiters in the encircling symbol are regarded as a part of the string value. Currently, the default encircling symbol of CDM is **"**.	No
	Using Escape Char	If you select **Yes**, the backslash (\) in the data row is used as an escape character. If you select **No**, the backslash (\) in the CSV file will not be escaped. CSV supports only the backslash (\) as the escape character.	Yes
	Use RE to Separate Fields	Whether to use regular expressions to separate fields. If you set this parameter to **Yes**, **Field Delimiter** becomes invalid. This parameter is displayed only when **File Format** is set to **CSV**.	Yes
	Regular Expression	This parameter is available only when **Using RE to separate fields** is set to **Yes**. Regular expression used to separate fields. For details about regular expressions, see **Regular Expressions for Separating Semi-structured Text**.	^(\d.*\d) (\w*) \[(.*) \] ([\w\.]*) (\w.*).*

Category	Parameter	Description	Example Value
	Use First Row as Header	This parameter is displayed only when **File Format** is set to **CSV**. When you migrate a CSV file to a table, CDM writes all data to the table by default. If you set this parameter to **Yes**, CDM uses the first N rows of the CSV file as the heading row and does not write the row to the destination table.	Yes
	Encoding Type	Encoding type, for example, **UTF-8** or **GBK**. You can set the encoding type for text files only. This parameter is invalid when **File Format** is set to **Binary**.	UTF-8
	Compression Format	The options are as follows: • **NONE**: Files in all formats can be transferred. • **GZIP**: Only files in gzip format can be transferred. • **ZIP**: Only files in Zip format can be transferred. • **TAR.GZ**: Files in TAR.GZ format are transferred.	NONE
	Compressed File Suffix	This parameter is displayed when **Compression Format** is not **NONE**. This parameter specifies the extension of the files to be decompressed. The decompression operation is performed only when the file name extension is used in a batch of files. Otherwise, files are transferred in the original format. If you enter ***** or leave the parameter blank, all files are decompressed.	*
	Start Job by Marker File	Whether to start a job by a marker file. A job is only started if there is a marker file for starting the job in the source path. If there is no marker file, the job will be suspended for a period of time specified by **Suspension Period**.	Yes
	File Separator	File separator. If you enter multiple file paths in **Source Directory/Files**, CDM uses the file separator to identify files. The default value is **|**.	|

Category	Parameter	Description	Example Value
	Marker File	Name of the marker file for starting a job. If you specify a marker file, the migration job is executed only when the marker file exists in the source path. The marker file will not be migrated.	ok.txt
	Suspension Period	Waiting period for a marker file. If you set **Start Job by Marker File** to **Yes** but there is no marker file in the source path, the job fails when the suspension period times out. If you set this parameter to **0** and there is no marker file in the source path, the job will fail immediately. Unit: second	10
	Filter Type	Only paths or files that meet the filtering conditions are transferred. The options are **None**, **Wildcard**, and **Regex**. For details, see **Incremental File Migration**.	None
	Directory Filter	If you set **Filter Type** to **Wildcard** or **Regex**, enter a wildcard character to filter paths. The paths that meet the filtering condition are migrated. You can configure multiple paths separated by commas (,). **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	*input,*out
	File Filter	If you set **Filter Type** to **Wildcard** or **Regex**, enter a wildcard character to filter paths. The files that meet the filtering condition are migrated. You can configure multiple files separated by commas (,). **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	*.csv
	Time Filter	If you select **Yes**, files are transferred based on their modification time.	Yes

Category	Parameter	Description	Example Value
	Minimum Timestamp	If you set **Time Filter** to **Yes**, you can specify a point in time for **Minimum Timestamp**, and then only the files modified at or after the specified time are transferred. The time format must be *yyyy-MM-dd HH:mm:ss*. This parameter can be set to a macro variable of date and time. For example, **${timestamp(dateformat(yyyy-MM-dd HH:mm:ss,-90,DAY))}** indicates that only files generated within the latest 90 days are migrated. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	2019-07-01 00:00:00
	Maximum Timestamp	If you set **Time Filter** to **Yes**, you can specify a point in time for **Maximum Timestamp**, and then only the files modified before the specified time are transferred. The time format must be *yyyy-MM-dd HH:mm:ss*. This parameter can be set to a macro variable of date and time. For example, **${timestamp(dateformat(yyyy-MM-dd HH:mm:ss))}** indicates that only the files whose modification time is earlier than the current time are migrated. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	2019-07-30 00:00:00
	Disregard Non-existent Path or File	If this parameter is set to **Yes**, the job can be successfully executed even if the source path does not exist.	No

Category	Parameter	Description	Example Value
	Marker File Type	This parameter is available only when **Start Job by Marker File** is set to **Yes**. • **MARK_DONE**: The migration job is executed only when the marker file exists in the source path. • **MARK_DOING**: The migration job is executed only when the marker file does not exist in the source path.	MARK_DOING
	Whether to skip empty lines	This parameter is available only when **File Format** is set to **CSV**. If a line is empty, it is skipped.	No
	null value	This parameter is available only when **File Format** is set to **Binary**. No string can be used to define a null value in text files. This parameter specifies the string to be identified as a null value.	No
	MD5 File Extension	This parameter is displayed only when **File Format** is set to **Binary**. This parameter is used to check whether the files extracted by CDM are consistent with source files. For details, see **MD5 Verification**.	.md5

4.6.3.7 From HTTP

If the source link of a job is an HTTP link, configure the source job parameters based on **Table 4-66**. Currently, data can only be exported from the HTTP URLs.

Table 4-66 Parameter description

Parameter	Description	Example Value
File URL	Use the GET method to obtain data from the HTTP/HTTPS URL. These connectors are used to read files with an HTTP/HTTPS URL, such as reading public files on the third-party object storage system and web disks.	https://bucket.obs.myhuaweicloud.com/object-key

Parameter	Description	Example Value
Pull List File	If this parameter is set to **Yes**, the system pulls the files corresponding to the URLs in the text file to be uploaded and stores them on OBS. The text file records the file paths on HDFS.	Yes
OBS Link of List File	Select an existing OBS link.	obs_link
OBS Bucket of entries files	Name of the OBS bucket that stores the text file	obs-cdm
Path/ Directory of entries files	Custom OBS directories that store the text file. Use slashes (/) to separate different directories.	test1
File Format	CDM supports **Binary** only, which indicates that files (even not in binary format) will be directly transferred.	Binary
Compression Format	Compression format of the source files. The options are as follows: ● **NONE**: Files in all formats can be transferred. ● **GZIP**: Only files in gzip format can be transferred. ● **ZIP**: Only files in Zip format can be transferred. ● **TAR.GZ**: Files in TAR.GZ format are transferred.	NONE
Compressed File Suffix	This parameter is displayed when **Compression Format** is not **NONE**. This parameter specifies the extension of the files to be decompressed. The decompression operation is performed only when the file name extension is used in a batch of files. Otherwise, files are transferred in the original format. If you enter ***** or leave the parameter blank, all files are decompressed.	*
File Separator	File separator. When multiple files are transferred, CDM uses the file separator to identify files. The default value is **|**. This parameter is not displayed if **Pull List File** is set to **Yes**.	|

Parameter	Description	Example Value
Query Parameter	• If you set this parameter to **Yes**, the name of the objects uploaded to OBS does not include the **query** parameter. • If you set this parameter to **No**, the name of the objects uploaded to OBS includes the **query** parameter.	No
Disregard Non-existent Path or File	If this is set to **Yes**, the job can be successfully executed even if the source path does not exist.	No
MD5 File Extension	This parameter is used to check whether the files extracted by CDM are consistent with source files. For details, see **MD5 Verification**.	.md5

4.6.3.8 From PostgreSQL/SQL Server

If the source link of a job is an RDS for PostgreSQL, RDS for SQL Server, PostgreSQL, or Microsoft SQL Server link, configure the source job parameters based on **Table 4-67**.

Table 4-67 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	Use SQL Statement	Whether you can use SQL statements to export data from a relational database	No

Category	Parameter	Description	Example Value

	SQL Statement	When **Use SQL Statement** is set to **Yes**, enter an SQL statement here. CDM exports data based on the SQL statement.
		NOTE
		• SQL statements can only be used to query data. Join and nesting are supported, but multiple query statements are not allowed, for example, **select * from table a; select * from table b**.
		• With statements are not supported.
		• Comments, such as **--** and **/***, are not supported.
		• Addition, deletion, and modification operations are not supported, including but not limited to the following:
		  • load data
		  • delete from
		  • alter table
		  • create table
		  • drop table
		  • into outfile
			select id,name from sqoop.user;

	Schema/ Tablespace	Name of the schema or tablespace from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting a schema or directly enter a schema or tablespace. If the desired schema or tablespace is not displayed, check whether the login account has the permissions required to query metadata. **NOTE** The parameter value can contain wildcard characters (*), which is used to export all databases whose names start with a certain prefix or end with a certain suffix. The examples are as follows: • **SCHEMA*** indicates that all databases whose names starting with **SCHEMA** are exported. • ***SCHEMA** indicates that all databases whose names ending with **SCHEMA** are exported. • ***SCHEMA*** indicates that all databases whose names containing **SCHEMA** are exported.	SCHEMA_E

Category	Parameter	Description	Example Value
	Table Name	Name of the table from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting the table or directly enter a table name. If the desired table is not displayed, check whether the table exists or whether the login account has the permission to query metadata. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*). **NOTE** The table name can contain wildcard characters (*), which is used to export all tables whose names start with a certain prefix or end with a certain suffix. The number and types of fields in the tables must be the same. The examples are as follows: • **table*** indicates that all tables whose names starting with **table** are exported. • ***table** indicates that all tables whose names ending with **table** are exported. • ***table*** indicates that all tables whose names containing **table** are exported.	table

Category	Parameter	Description	Example Value
Advanced attributes	Partition Column	This parameter is displayed when **Use SQL Statement** is set to **No**, indicating that a field used to split data during data extraction. CDM splits a job into multiple tasks based on this field and executes the tasks concurrently. Fields with data distributed evenly are used, such as the sequential number field. Click the icon next to the text box to go to the page for selecting a field or directly enter a field. **NOTE** The following types of partition columns are supported: TINYINT, SMALLINT, INTEGER, BIGINT, REAL, FLOAT, DOUBLE, NUMERIC, DECIMAL, BIT, BOOLEAN, DATE, TIME, and TIMESTAMP. It is recommended that the partition column have an index.	id
	Where Clause	WHERE clause used to specify the data extraction range. This parameter is displayed when **Use SQL Statement** is set to **No**. If this parameter is not set, the entire table is extracted. You can set a date macro variable to extract data generated on a specific date. For details, see **Incremental Migration of Relational Databases**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	DS='${dateformat(yyyy-MM-dd,-1,DAY)}'
	Null in Partition Column	Whether the partition column can contain null values	Yes

Category	Parameter	Description	Example Value
	Extract by Partition	Data can be extracted from each partition in the partitioned table. If this function is enabled, you can configure **Table Partition** to specify specific table partitions from which data is extracted. • This function does not support non-partitioned tables. • This parameter can be configured only when the migration source is a PostgreSQL database. • The database user must have the **SELECT** permission on the system views **dba_tab_partitions** and **dba_tab_subpartitions**.	No
	Split Job	If this parameter is set to **Yes**, the job is split into multiple subjobs based on the value of **Job Split Field**, and the subjobs are executed concurrently. **NOTE** This parameter and parameters *Job Split Field*, *Minimum Split Field Value*, *Maximum Split Field Value*, and *Number of subjobs* are available only when the destination link is a DLI or Hive link.	Yes
	Job Split Field	Field used to split a job into multiple subjobs for concurrent execution. This parameter is available when **Split Job** is set to **Yes**.	-
	Minimum Split Field Value	Minimum value of **Job Split Field** during data extraction. This parameter is available when **Split Job** is set to **Yes**.	-
	Maximum Split Field Value	Maximum value of **Job Split Field** during data extraction. This parameter is available when **Split Job** is set to **Yes**.	-
	Number of subjobs	Number of subjobs split from a job for concurrent execution based on the data range specified by the minimum and maximum values of **Job Split Field**. This parameter is available when **Split Job** is set to **Yes**.	-

4.6.3.9 From DWS

If the source link of a job is a **DWS link**, configure the source job parameters based on **Table 4-68**.

Table 4-68 Parameter description

Type	Parameter	Description	Example Value
Basic parameters	Use SQL Statement	Whether you can use SQL statements to export data from a relational database	No

	SQL Statement	When **Use SQL Statement** is set to **Yes**, enter an SQL statement here. CDM exports data based on the SQL statement.
		NOTE
		• SQL statements can only be used to query data. Join and nesting are supported, but multiple query statements are not allowed, for example, **select * from table a; select * from table b**.
		• With statements are not supported.
		• Comments, such as **--** and **/***, are not supported.
		• Addition, deletion, and modification operations are not supported, including but not limited to the following:
		  • load data
		  • delete from
		  • alter table
		  • create table
		  • drop table
		  • into outfile
			select id,name from sqoop.user;

Type	Parameter	Description	Example Value
	Schema/Tablespace	Name of the schema or tablespace from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting a schema or directly enter a schema or tablespace. If the desired schema or tablespace is not displayed, check whether the login account has the permissions required to query metadata. **NOTE** The parameter value can contain wildcard characters (*), which is used to export all databases whose names start with a certain prefix or end with a certain suffix. Examples: • **SCHEMA*** indicates that all databases whose names starting with **SCHEMA** are exported. • ***SCHEMA** indicates that all databases whose names ending with **SCHEMA** are exported. • ***SCHEMA*** indicates that all databases whose names containing **SCHEMA** are exported.	SCHEMA_E

Type	Parameter	Description	Example Value
	Table Name	Name of the table from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting the table or directly enter a table name. If the desired table is not displayed, check whether the table exists or whether the login account has the permission to query metadata. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*). **NOTE** The table name can contain wildcard characters (*), which is used to export all tables whose names start with a certain prefix or end with a certain suffix. The number and types of fields in the tables must be the same. The examples are as follows: • **table*** indicates that all tables whose names starting with **table** are exported. • ***table** indicates that all tables whose names ending with **table** are exported. • ***table*** indicates that all tables whose names containing **table** are exported.	table

Type	Parameter	Description	Example Value
Advanced attributes	WHERE Clause	WHERE clause used to specify the data extraction range. This parameter is displayed when **Use SQL Statement** is set to **No**. If this parameter is not set, the entire table is extracted. You can set a date macro variable to extract data generated on a specific date. For details, see **Incremental Migration of Relational Databases**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	DS='${dateformat(yyyy-MM-dd,-1,DAY)}'
	Partition Column	This parameter is displayed when **Use SQL Statement** is set to **No**, indicating a field used to split data during data extraction. CDM splits a job into multiple tasks based on this field and executes the tasks concurrently. Fields with data distributed evenly are used, such as the sequential number field. Click the icon next to the text box to go to the page for selecting a field or directly enter a field. **NOTE** The following types of partition columns are supported: TINYINT, SMALLINT, INTEGER, BIGINT, REAL, FLOAT, DOUBLE, NUMERIC, DECIMAL, BIT, BOOLEAN, DATE, TIME, and TIMESTAMP. It is recommended that the partition column have an index.	id
	Null in Partition Column	Whether the partition column can contain null values	Yes
	Split Job	If this parameter is set to **Yes**, the job is split into multiple subjobs based on the value of **Job Split Field**, and the subjobs are executed concurrently. **NOTE** This parameter and parameters *Job Split Field*, *Minimum Split Field Value*, *Maximum Split Field Value*, and *Number of subjobs* are available only when the destination link is a DLI or Hive link.	Yes

Type	Parameter	Description	Example Value
	Job Split Field	Field used to split a job into multiple subjobs for concurrent execution. This parameter is available when **Split Job** is set to **Yes**.	-
	Minimum Split Field Value	Minimum value of **Job Split Field** during data extraction. This parameter is available when **Split Job** is set to **Yes**.	-
	Maximum Split Field Value	Maximum value of **Job Split Field** during data extraction. This parameter is available when **Split Job** is set to **Yes**.	-
	Number of subjobs	Number of subjobs split from a job for concurrent execution based on the data range specified by the minimum and maximum values of **Job Split Field**. This parameter is available when **Split Job** is set to **Yes**.	-

4.6.3.10 From SAP HANA

Table 4-69 lists the job parameters when the source link is a SAP HANA link.

Table 4-69 Parameter description

Type	Parameter	Description	Example Value
Basic parameters	Use SQL Statement	Whether you can use SQL statements to export data from a relational database	No

Type	Parameter	Description	Example Value

	SQL Statement	When **Use SQL Statement** is set to **Yes**, enter an SQL statement here. CDM exports data based on the SQL statement.
		NOTE
		• SQL statements can only be used to query data. Join and nesting are supported, but multiple query statements are not allowed, for example, **select * from table a; select * from table b**.
		• With statements are not supported.
		• Comments, such as **--** and **/***, are not supported.
		• Addition, deletion, and modification operations are not supported, including but not limited to the following:
		  • load data
		  • delete from
		  • alter table
		  • create table
		  • drop table
		  • into outfile
			select id,name from sqoop.user;

	Schema/ Tablespace	Name of the schema or tablespace from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting a schema or directly enter a schema or tablespace. If the desired schema or tablespace is not displayed, check whether the login account has the permissions required to query metadata. **NOTE** The parameter value can contain wildcard characters (*), which is used to export all databases whose names start with a certain prefix or end with a certain suffix. Examples: • **SCHEMA*** indicates that all databases whose names starting with **SCHEMA** are exported. • ***SCHEMA** indicates that all databases whose names ending with **SCHEMA** are exported. • ***SCHEMA*** indicates that all databases whose names containing **SCHEMA** are exported.	SCHEMA_E

Type	Parameter	Description	Example Value
	Table Name	Name of the table from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting the table or directly enter a table name. If the desired table is not displayed, check whether the table exists or whether the login account has the permission to query metadata. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*). **NOTE** The table name can contain wildcard characters (*), which is used to export all tables whose names start with a certain prefix or end with a certain suffix. The number and types of fields in the tables must be the same. The examples are as follows: • **table*** indicates that all tables whose names starting with **table** are exported. • ***table** indicates that all tables whose names ending with **table** are exported. • ***table*** indicates that all tables whose names containing **table** are exported.	table

Type	Parameter	Description	Example Value
Advanced attributes	WHERE Clause	WHERE clause used to specify the data extraction range. This parameter is displayed when **Use SQL Statement** is set to **No**. If this parameter is not set, the entire table is extracted. You can set a date macro variable to extract data generated on a specific date. For details, see **Incremental Migration of Relational Databases**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	DS='${dateformat(yyyy-MM-dd,-1,DAY)}'
	Partition Column	This parameter is displayed when **Use SQL Statement** is set to **No**, indicating a field used to split data during data extraction. CDM splits a job into multiple tasks based on this field and executes the tasks concurrently. Fields with data distributed evenly are used, such as the sequential number field. Click the icon next to the text box to go to the page for selecting a field or directly enter a field. **NOTE** The following types of partition columns are supported: TINYINT, SMALLINT, INTEGER, BIGINT, REAL, FLOAT, DOUBLE, NUMERIC, DECIMAL, BIT, BOOLEAN, DATE, TIME, and TIMESTAMP. It is recommended that the partition column have an index.	id

4.6.3.11 From MySQL

If the source link of a job is an **RDS for MySQL or MySQL link**, configure the source job parameters based on **Table 4-70**.

Table 4-70 Parameter description

Parameter	Description	Example Value
Use SQL Statement	Whether you can use SQL statements to export data from a relational database	No

Parameter	Description	Example Value

SQL Statement	When **Use SQL Statement** is set to **Yes**, enter an SQL statement here. CDM exports data based on the SQL statement.
	NOTE
	• SQL statements can only be used to query data. Join and nesting are supported, but multiple query statements are not allowed, for example, **select * from table a; select * from table b**.
	• With statements are not supported.
	• Comments, such as **--** and **/***, are not supported.
	• Addition, deletion, and modification operations are not supported, including but not limited to the following:
	  • load data
	  • delete from
	  • alter table
	  • create table
	  • drop table
	  • into outfile
		select id,name from sqoop.user;

Schema/ Tablespace	Name of the schema or tablespace from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting a schema or directly enter a schema or tablespace. If the desired schema or tablespace is not displayed, check whether the login account has the permissions required to query metadata. This parameter can be set to a regular expression to export all databases that meet the rule.	SCHEMA_E

Parameter	Description	Example Value
Table Name	Name of the table from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting the table or directly enter a table name. If the desired table is not displayed, check whether the table exists or whether the login account has the permission to query metadata. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. This parameter can be set to a regular expression to export all databases that meet the rule. For example, if **Table Name** is set to *user_[0-9]{1,2}*, tables from **user_0** to **user_9** and from **user_00** to **user_99** are matched. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	table
Partition Column	This parameter is displayed when **Use SQL Statement** is set to **No**, indicating a field used to split data during data extraction. CDM splits a job into multiple tasks based on this field and executes the tasks concurrently. Fields with data distributed evenly are used, such as the sequential number field. Click the icon next to the text box to go to the page for selecting a field or directly enter a field. **NOTE** The following types of partition columns are supported: TINYINT, SMALLINT, INTEGER, BIGINT, REAL, FLOAT, DOUBLE, NUMERIC, DECIMAL, BIT, BOOLEAN, DATE, TIME, and TIMESTAMP. It is recommended that the partition column have an index.	id

Parameter	Description	Example Value
Where Clause	WHERE clause used to specify the data extraction range. This parameter is displayed when **Use SQL Statement** is set to **No**. If this parameter is not set, the entire table is extracted. You can set a date macro variable to extract data generated on a specific date. For details, see **Incremental Migration of Relational Databases**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	DS='${dateformat(yyyy-MM-dd,-1,DAY)}'
Null in Partition Column	Whether the partition column can contain null values	Yes
Split Job	If this parameter is set to **Yes**, the job is split into multiple subjobs based on the value of **Job Split Field**, and the subjobs are executed concurrently. **NOTE** This parameter and parameters *Job Split Field*, *Minimum Split Field Value*, *Maximum Split Field Value*, and *Number of subjobs* are available only when the destination link is a DLI or Hive link.	Yes
Job Split Field	Field used to split a job into multiple subjobs for concurrent execution. This parameter is available when **Split Job** is set to **Yes**.	-
Minimum Split Field Value	Minimum value of **Job Split Field** during data extraction. This parameter is available when **Split Job** is set to **Yes**.	-
Maximum Split Field Value	Maximum value of **Job Split Field** during data extraction. This parameter is available when **Split Job** is set to **Yes**.	-
Number of subjobs	Number of subjobs split from a job for concurrent execution based on the data range specified by the minimum and maximum values of **Job Split Field**. This parameter is available when **Split Job** is set to **Yes**.	-

Parameter	Description	Example Value
Extract by Partition	When data is exported from a MySQL database, data can be extracted from each partition in the partitioned table. If this function is enabled, you can configure **Table Partition** to specify specific MySQL table partitions from which data is extracted. • This function does not support non-partitioned tables. • The database user must have the **SELECT** permission on the system views **dba_tab_partitions** and **dba_tab_subpartitions**.	No

4.6.3.12 From Oracle

If the source link of a job is an **Oracle link**, configure the source job parameters based on **Table 4-71**.

Table 4-71 Parameter description

Parameter	Description	Example Value
Use SQL Statement	Whether you can use SQL statements to export data from a relational database	No

SQL Statement	When **Use SQL Statement** is set to **Yes**, enter an SQL statement here. CDM exports data based on the SQL statement.
	NOTE
	• SQL statements can only be used to query data. Join and nesting are supported, but multiple query statements are not allowed, for example, **select * from table a; select * from table b**.
	• With statements are not supported.
	• Comments, such as **--** and **/***, are not supported.
	• Addition, deletion, and modification operations are not supported, including but not limited to the following:
	  • load data
	  • delete from
	  • alter table
	  • create table
	  • drop table
	  • into outfile
		select id,name from sqoop.user;

Parameter	Description	Example Value
Schema/ Tablespace	Name of the schema or tablespace from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting a schema or directly enter a schema or tablespace. If the desired schema or tablespace is not displayed, check whether the login account has the permissions required to query metadata. **NOTE** The parameter value can contain wildcard characters (*), which is used to export all databases whose names start with a certain prefix or end with a certain suffix. For example: • **SCHEMA*** indicates that all databases whose names starting with **SCHEMA** are exported. • ***SCHEMA** indicates that all databases whose names ending with **SCHEMA** are exported. • ***SCHEMA*** indicates that all databases whose names containing **SCHEMA** are exported.	SCHEMA_E

Parameter	Description	Example Value
Table Name	Name of the table from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting the table or directly enter a table name. If the desired table is not displayed, check whether the table exists or whether the login account has the permission to query metadata. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*). **NOTE** The table name can contain wildcard characters (*), which is used to export all tables whose names start with a certain prefix or end with a certain suffix. The number and types of fields in the tables must be the same. The examples are as follows: • **table*** indicates that all tables whose names starting with **table** are exported. • ***table** indicates that all tables whose names ending with **table** are exported. • ***table*** indicates that all tables whose names containing **table** are exported.	table

Parameter	Description	Example Value
Partition Column	This parameter is displayed when **Extract by Partition** is set to **No**, indicating a field used to split data during data extraction. CDM splits a job into multiple tasks based on this field and executes the tasks concurrently. Fields with data distributed evenly are used, such as the sequential number field. Click the icon next to the text box to go to the page for selecting a field or directly enter a field. **NOTE** The following types of partition columns are supported: TINYINT, SMALLINT, INTEGER, BIGINT, REAL, FLOAT, DOUBLE, NUMERIC, DECIMAL, BIT, BOOLEAN, DATE, TIME, and TIMESTAMP. It is recommended that the partition column have an index.	id
Where Clause	WHERE clause used to specify the data extraction range. This parameter is displayed when **Use SQL Statement** is set to **No**. If this parameter is not set, the entire table is extracted. You can set a date macro variable to extract data generated on a specific date. For details, see **Incremental Migration of Relational Databases**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	DS='${dateformat(yyyy-MM-dd,-1,DAY)}'
Null in Partition Column	Whether the partition field can contain null values. This parameter is displayed when **Extract by Partition** is set to **No**.	Yes
Extract by Partition	When data is exported from an Oracle database, data can be extracted from each partition in the partitioned table. If this function is enabled, you can configure **Table Partition** to specify specific Oracle table partitions from which data is extracted. • This function does not support non-partitioned tables. • The database user must have the **SELECT** permission on the system views **dba_tab_partitions** and **dba_tab_subpartitions**.	No

Parameter	Description	Example Value
Table Partition	Oracle table partition from which data is migrated. Separate multiple partitions with ampersands (&). If you do not set this parameter, all partitions will be migrated. If there is a subpartition, enter the partition in the *Partition.Subpartition* format, for example, **P2.SUBP1**.	P0&P1&P2.SUBP1&P2.SUBP3
Split Job	If this parameter is set to **Yes**, the job is split into multiple subjobs based on the value of **Job Split Field**, and the subjobs are executed concurrently. **NOTE** This parameter and parameters *Job Split Field*, *Minimum Split Field Value*, *Maximum Split Field Value*, and *Number of subjobs* are available only when the destination link is a DLI or Hive link.	Yes
Job Split Field	Field used to split a job into multiple subjobs for concurrent execution. This parameter is available when **Split Job** is set to **Yes**.	-
Minimum Split Field Value	Minimum value of **Job Split Field** during data extraction. This parameter is available when **Split Job** is set to **Yes**.	-
Maximum Split Field Value	Maximum value of **Job Split Field** during data extraction. This parameter is available when **Split Job** is set to **Yes**.	-
Number of subjobs	Number of subjobs split from a job for concurrent execution based on the data range specified by the minimum and maximum values of **Job Split Field**. This parameter is available when **Split Job** is set to **Yes**.	-

NOTE

When an Oracle database is the migration source, if **Partitioning Field** or **Extract by Partition** is not configured, CDM automatically uses the ROWIDs to partition data.

4.6.3.13 From a Database Shard

If the source link of a job is a **database shard link**, configure the source job parameters based on **Table 4-72**.

Table 4-72 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	Schema/Tablespace	Indicates the name of the schema or tablespace from which data is to be extracted. Click the icon next to the text box to go to the page for selecting a schema or tablespace. During a sharded link job, the tablespace corresponding to the first backend link is displayed by default. You can also enter a schema or tablespace name. If the desired schema or tablespace is not displayed, check whether the login account has the permissions required to query metadata. This parameter can be set to a regular expression to export all databases that meet the rule. For example, if **Table Name** is set to *user_[0-9]{1,2}*, tables from **user_0** to **user_9** and from **user_00** to **user_99** are matched.	SCHEMA_E

Category	Parameter	Description	Example Value
	Table Name	Indicates the name of the table from which data is to be extracted. Click the icon next to the text box to go to the page for selecting the table or directly enter a table name. If the desired table is not displayed, check whether the table exists or whether the login account has the permission to query metadata. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. This parameter can be set to a regular expression to export all databases that meet the rule. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	table
Advanced attributes	WHERE Clause	Specifies the data extraction range. If this parameter is not set, the entire table is extracted. You can set a date macro variable to extract data generated on a specific date. For details, see **Incremental Migration of Relational Databases**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	DS='${dateformat(yyyy-MM-dd,-1,DAY)}'

NOTE

- If the **Source Link Name** is the backend link of the sharded link, the job is a common MySQL job.
- When creating a job whose source end is a sharded link, you can add a custom field with the sample value of **${custom(host)}** to the source field during field mapping. This field is used to view the data source of the table after the data of multiple tables across databases is migrated to the same table. The following sample values are supported:
  - ${custom(host)}
  - ${custom(database)}
  - ${custom(fromLinkName)}
  - ${custom(schemaName)}
  - ${custom(tableName)}

4.6.3.14 From MongoDB/DDS

When you migrate MongoDB or DDS data, CDM reads the first row of the collection as an example of the field list. If the first row of data does not contain all fields of the collection, you need to manually add fields.

If the source link of a job is a **MongoDB link**, that is, if data is exported from an on-premises MongoDB or DDS, configure the source job parameters based on **Table 4-73**.

Table 4-73 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	Database Name	Name of the database from which data will be migrated	mongodb
	Collection Name	Collection name, similar to the table name of a relational database. Click the icon next to the text box to go to the page for selecting the collection or directly enter a collection name. If the desired table is not displayed, check whether the table exists or whether the login account has the permission to query metadata.	COLLECTION

Category	Parameter	Description	Example Value
Advanced attributes	Filter Condition	Conditions for filtering documents. CDM migrates only the data that meets the filter conditions. The examples are as follows: 1. Filter by expression: **{'last_name': 'Smith'}** indicates that all files whose **last_name** value is **Smith** are queried. 2. Filter by parameter: **{ x : "john" }, { z : 1 }** indicates that all **z** fields whose **x** is **john** are queried. 3. Filter by condition: **{ "field" : { $gt: 5 } }** indicates that the **field** values greater than 5 are queried. 4. Filter by time macro: {"ts":{$gte:ISODate("${dateformat(yyyy-MM-dd'T'HH:mm:ss.SSS'Z',-1,HOUR)}")}} indicates that the values greater than those after time macro conversion in the **ts** field are queried.	{'last_name': 'Smith'}

4.6.3.15 From Redis

The Redis service of the third-party cloud cannot serve as the migration source. However, the Redis set up in the on-premises data center or on the ECS can be the migration source and destination.

If the source link of a job is an on-premises Redis link, configure the source job parameters based on **Table 4-74**.

Table 4-74 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	Redis Key Prefix	Key prefix, which is similar to the table name of a relational database	TABLE
	Value Storage Type	The options are as follows: • **String**: without column name, such as **value1,value2** • **Hash**: with column name, such as **column1=value1,column2=value2**	String

Category	Parameter	Description	Example Value
Advanced attributes	Key Delimiter	Character used to separate table names and column names of a relational database	_
	Value Delimiter	Character used to separate columns when the storage type is string	;
	Same Field	This parameter is displayed when **Value Storage Type** is set to **Hash**. The hash key contains the same field.	Yes

4.6.3.16 From DIS

The data in the message body is a record in CSV format that supports multiple delimiters. Messages cannot be parsed in binary or other formats.

If the source link of a job is a **DIS link**, configure the source job parameters based on **Table 4-75**.

Table 4-75 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	DIS Stream	DIS stream name	dis
	Permanent Running	Whether a job runs permanently. If a job is set to run for a long time, the job will fail if the DIS system is interrupted.	Yes
	DIS Partition ID	ID of the DIS partition. You can enter multiple partition IDs separated by commas (,).	0,1,2
	Offset	Initial offset when data is pulled from DIS • **Latest**: Maximum offset, indicating that the latest data will be extracted. • **From last stop**: Data read will start from which the last read ended. • **Earliest**: Minimum offset, indicating that the earliest data will be extracted.	Latest
	Application Name	Unique identifier of the consumer application to be used. If no application exists, CDM creates one automatically.	cdm

Category	Parameter	Description	Example Value
	Data Format	Format used for parsing data. The options are as follows: • **Binary**: Data is transferred directly. It is not converted to another format. This setting is suitable for file migration. • **CSV**: Source data will be migrated after being converted in CSV format. • **JSON**: Source data will be migrated after being converted in JSON format.	Binary
	Field Delimiter	This parameter is displayed when **Data Format** is set to **CSV**. The default value is comma (,). To set the **Tab** key as the delimiter, set this parameter to **\t**.	,
	Record Delimiter	This parameter is displayed when **Data Format** is set to **CSV** or **JSON**. It is used to separate each two records.	,
Advanced attributes	Max. Poll Records	(Optional) Maximum number of records per poll	100

4.6.3.17 From Kafka/DMS Kafka

If the source link of a job is a **Kafka link** or **DMS Kafka link**, configure the source job parameters based on **Table 4-76**.

Table 4-76 Parameter description

Type	Parameter	Description	Example Value
Basic parameters	Topics	One or more topics can be entered.	est1,est2

Type	Parameter	Description	Example Value
	Data Format	Format used for parsing data. The options are as follows: • **Binary**: Data is transferred directly. It is not converted to another format. This setting is suitable for file migration. • **CSV**: Source data will be migrated after being converted in CSV format. • **JSON**: Source data will be migrated after being converted in JSON format. • **CDC (DRS)**: Source data will be migrated after being converted in DRS format. • **CDC (JSON)**: Source data will be migrated after being converted in JSON format. • **CDC (DRS_AVRO)**: Source data will be migrated after being converted in DRS_AVRO format. • **CDC (DRS_JSON)**: Source data will be migrated after being converted in DRS_JSON format.	Binary
	Offset	Initial offset parameter • **Latest**: Maximum offset, indicating that the latest data will be extracted. • **Earliest**: Minimum offset, indicating that the earliest data will be extracted. • **Submitted**: data that has been submitted • **Time Range**: data within a specified time range	Latest
	Data Extraction Timeout Duration	Maximum duration (minutes) of data extraction. For example, a job scheduled daily needs a sufficient duration to extract the data generated by the topic every day.	60
	Suspension Period	If the value is set to 60 and no data is returned within 60s after the consumer requests data extraction from Kafka (generally because all the data in the topic has been read or the network or Kafka cluster is unavailable), the task will stop immediately. Otherwise, the system will retry reading data.	60
	Consumer Group ID	Consumer group ID If you export data from DMS Kafka, enter any value for Kafka Platinum but a valid consumer group ID for Kafka Basic.	sumer-group

Type	Parameter	Description	Example Value
	Start Time	This parameter is required when **Offset** is set to **Time Range**. It specifies the start time for pulling data, including the data at the specified time point.	2020-12-20 12:00:00
	End Time	This parameter is required when **Offset** is set to **Time Range**. It specifies the end time for pulling data, excluding the data at the specified time point.	2020-12-20 20:00:00
	Field Delimiter	This parameter is required when **Data Format** is set to **CSV**. The default value is space. To set the **Tab** key as the delimiter, set this parameter to **\t**.	,
	Record Delimiter	This parameter is required when **Data Format** is set to **CSV** or **JSON**. The default value is space. To set the **Tab** key as the delimiter, set this parameter to **\t**.	,
Advanced parameters	UseConfigFile	This parameter is required when **Data Format** is set to **CDC**. It is used to configure OBS files.	No
	OBS Link	Select an OBS link.	obs_link
	OBS Bucket	Select an OBS bucket.	obs_test
	Config File	Select the OBS configuration file.	/obs/config.csv
	Max. Poll Records	(Optional) Maximum number of records per poll	100
	Max. Poll Interval	(Optional) Maximum interval between polls (seconds)	100
	Notice Topic	Topic for sending notification data. If the data format is CDC, the notification content is the names of the generated files.	notice

4.6.3.18 From Elasticsearch or CSS

If the source link of a job is a link described in **Link to Elasticsearch** or **Link to CSS**, configure the source job parameters based on **Table 4-77**.

Table 4-77 Job parameters when Elasticsearch or CSS is the source

Category	Parameter	Description	Example Value
Basic parameters	Index	Elasticsearch index, which is similar to the name of a relational database. The index name can contain only lowercase letters.	index
	Type	Elasticsearch type, which is similar to the table name of a relational database. The type name can contain only lowercase letters. **NOTE** Elasticsearch 7.x and later versions do not support custom types. Instead, only the **_doc** type can be used. In this case, this parameter does not take effect even if it is set.	_doc
Advanced attributes	Split Nested Field	(Optional) Whether to split the JSON content of the nested fields. For example, **a:{ b:{ c:1, d:{ e:2, f:3 } } }** can be split into **a.b.c**, **a.b.d.e**, and **a.b.d.f**.	No

Category	Parameter	Description	Example Value
	Filter Conditions	(Optional) CDM migrates only the data that meets the filter conditions. • Currently, only the query string (q syntax) of Elasticsearch can be used to filter source data. The q syntax is used in the following way: – In exact match, the ***column*:*data*** format is used to match and filter data. ***column*** indicates the field name, and ***data*** indicates the query condition, for example, **last_name:Smith**. In addition, if ***data*** is a string containing spaces, it must be enclosed in double quotation marks. If ***column*** is not specified, all fields will be matched by ***data***. – Multiple query conditions can be combined with connection words. The format is ***column1*:*data1* AND *column2*:*data2***. The connection words can be **AND**, **OR**, or **NOT**. They must be in uppercase, and there must be a space before and after each connection word. Example: **first_name:Alec AND last_name:John** – In range matching, you can directly use a condition expression to filter data. The expression is in ***column*:>*data*** format. The operator can be >, >=, <, or <=. An example is **time:>=1636905600000 AND time:<1637078400000**. It can also be used together with a macro variable of date and time, for example, **createTime:>=${timestamp(dateformat(yyyyMMdd,-1,DAY))} AND createTime:< ${timestamp(dateformat(yyyyMMdd))}**. – In range matching, you can also use the range syntax to filter data. The format is ***column*:{*data1* TO *data2*}**. **{** and **}** indicate that a value is not included. **[** and **]** indicate that a	last_name:Smith

Category	Parameter	Description	Example Value
		value is included. **TO** must be capitalized, and there must be a space before and after it. * indicates all data. For example, **time:{1636992000000 TO *]** filters out all the data greater than 1636992000000 in the **time** field. It can also be used together with a macro variable of date and time, for example, **createTime:[${timestamp(dateformat(yyyyMMdd,-1,DAY))} TO ${timestamp(dateformat(yyyyMMdd))}}**. • Source data cannot be filtered using the query domain-specific language (DSL) of Elasticsearch.	
	Extract Meta-field	Whether to extract index meta-fields. For example, _index, _type, _id, and _score.	Yes
	Page size	Elasticsearch page size	1000
	ScrollId Time Out	During a scroll query using Elasticsearch, a **scroll_id** is recorded. When the query times out or is complete, the recorded **srcoll_id** will be cleared. You can set this parameter to specify the timeout duration.	5

4.6.3.19 From OpenTSDB

If the source link of a job is a **CloudTable OpenTSDB link**, configure the source job parameters based on **Table 4-78**.

Table 4-78 Parameter description

Parameter	Description	Example Value
Start Time	Start time of the query. The value is a character string or timestamp in the format of *yyyyMMddHHmmdd*.	20180920145505
End Time	(Optional) End time of the query. The value is a string or timestamp in the format of *yyyyMMddHHmmdd*.	20180921145505

Parameter	Description	Example Value
Metric	Metric of the data to be migrated. You can specify a metric or select an existing metric in OpenTSDB.	city.temp
Aggregate Function	Aggregate function	sum
Tag	(Optional) If you specify a tag, only the tagged data will be migrated.	tagk1:tagv1,tagk2:tagv2

4.6.3.20 From MRS Hudi

If the source link of a job is an **MRS Hudi link**, configure the source job parameters based on **Table 4-79**.

Table 4-79 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	Source Link Name	MRS Hudi link	hudi_from_cdm
	Database Name	Database name. Click the icon next to the text box. The dialog box for selecting the database is displayed.	default
	Table Name	Hudi table name. Click the icon next to the text box. The dialog box for selecting the table is displayed. You can set a macro variable of date and time, and a path name can contain multiple macro variables. You can use macro variables of date and time in a scheduled job to synchronize incremental data periodically. For details, see **Using Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	TBL_E

Category	Parameter	Description	Example Value
Advanced attributes	Where Clause	This parameter indicates the where clause to be extracted. If this parameter is not set, the entire table is extracted. If the table to be migrated does not contain the fields specified by the where clause, the migration will fail. You can set a macro variable of date and time to extract the data generated on a specific date. For details, see **Incremental Migration of Relational Databases**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	age > 18 and age <= 60

4.6.3.21 From MRS ClickHouse

If the source link of a job is an **MRS ClickHouse link**, configure the source job parameters based on **Table 4-80**.

Table 4-80 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	Source Link Name	MRS ClickHouse link	ck_from_cdm
	Schema/ Tablespace	Click the icon next to the text box to go to the page for selecting a schema or directly enter a schema or tablespace. If the desired schema or tablespace is not displayed, check whether the login account has the permissions required to query metadata. **NOTE** This parameter can be set to a regular expression to export all databases that meet the rule.	default

Category	Parameter	Description	Example Value
	Table Name	Click the icon next to the text box to go to the page for selecting the table or directly enter a table name. If the desired table is not displayed, check whether the table exists or whether the login account has the permission to query metadata. **NOTE** This parameter can be set to a regular expression to export all databases that meet the rule.	TBL_E
Advanced attributes	WHERE Clause	This parameter indicates the WHERE clause to be extracted. If this parameter is not set, the entire table is extracted. If the table to be migrated does not contain the fields specified by the WHERE clause, the migration will fail. You can set a date macro variable to extract data generated on a specific date. For details, see **Incremental Migration of Relational Databases**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	age > 18 and age <= 60

4.6.3.22 From a ShenTong Database

If the source link of a job is a ShenTong database link, configure the source job parameters based on **Table 4-81**.

Table 4-81 Parameter description

Type	Parameter	Description	Example Value
Basic parameters	Use SQL Statement	Whether you can use SQL statements to export data from a relational database	No

Type	Parameter	Description	Example Value

	SQL Statement	When **Use SQL Statement** is set to **Yes**, enter an SQL statement here. CDM exports data based on the SQL statement.
		NOTE
		• SQL statements can only be used to query data. Join and nesting are supported, but multiple query statements are not allowed, for example, **select * from table a; select * from table b**.
		• With statements are not supported.
		• Comments, such as **--** and **/***, are not supported.
		• Addition, deletion, and modification operations are not supported, including but not limited to the following:
		  • load data
		  • delete from
		  • alter table
		  • create table
		  • drop table
		  • into outfile
		• Quote characters take effect only for SQL statements generated in the database table configuration, and cannot be added to custom SQL statements.
			select id,name from sqoop.user;

	Schema/ Tablespace	Name of the schema or tablespace from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting a schema or directly enter a schema or tablespace. If the desired schema or tablespace is not displayed, check whether the login account has the permissions required to query metadata. **NOTE** The parameter value can contain wildcard characters (*), which is used to export all databases whose names start with a certain prefix or end with a certain suffix. For example: • **SCHEMA*** indicates that all databases whose names starting with **SCHEMA** are exported. • ***SCHEMA** indicates that all databases whose names ending with **SCHEMA** are exported. • ***SCHEMA*** indicates that all databases whose names containing **SCHEMA** are exported.	SCHEMA_E

Type	Parameter	Description	Example Value
	Table Name	Name of the table from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting the table or directly enter a table name. If the desired table is not displayed, check whether the table exists or whether the login account has the permission to query metadata. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*). **NOTE** The table name can contain wildcard characters (*), which is used to export all tables whose names start with a certain prefix or end with a certain suffix. The number and types of fields in the tables must be the same. The examples are as follows: • **table*** indicates that all tables whose names starting with **table** are exported. • ***table** indicates that all tables whose names ending with **table** are exported. • ***table*** indicates that all tables whose names containing **table** are exported.	table

Type	Parameter	Description	Example Value
Advanced attributes	Partition Column	This parameter is displayed when **Use SQL Statement** is set to **No**, indicating that a field used to split data during data extraction. CDM splits a job into multiple tasks based on this field and executes the tasks concurrently. Fields with data distributed evenly are used, such as the sequential number field. Click the icon next to the text box to go to the page for selecting a field or directly enter a field. **NOTE** The following types of partition columns are supported: TINYINT, SMALLINT, INTEGER, BIGINT, REAL, FLOAT, DOUBLE, NUMERIC, DECIMAL, BIT, BOOLEAN, DATE, TIME, and TIMESTAMP. It is recommended that the partition column have an index.	id
	WHERE Clause	WHERE clause used to specify the data extraction range. This parameter is displayed when **Use SQL Statement** is set to **No**. If this parameter is not set, the entire table is extracted. You can set a date macro variable to extract data generated on a specific date. For details, see **Incremental Migration of Relational Databases**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	DS='${dateformat(yyyy-MM-dd,-1,DAY)}'
	Null in Partition Column	Whether the partition column can contain null values	Yes

4.6.3.23 From a Dameng Database

If the source link of a job is a Dameng database link, configure the source job parameters based on **Table 4-82**.

Table 4-82 Parameter description

Type	Parameter	Description	Example Value
Basic parameters	Use SQL Statement	Whether you can use SQL statements to export data from a relational database	No

	SQL Statement	When **Use SQL Statement** is set to **Yes**, enter an SQL statement here. CDM exports data based on the SQL statement.
		NOTE
		• SQL statements can only be used to query data. Join and nesting are supported, but multiple query statements are not allowed, for example, **select * from table a; select * from table b**.
		• With statements are not supported.
		• Comments, such as **--** and **/***, are not supported.
		• Addition, deletion, and modification operations are not supported, including but not limited to the following:
		  • load data
		  • delete from
		  • alter table
		  • create table
		  • drop table
		  • into outfile
			select id,name from sqoop.user;

	Schema/ Tablespace	Name of the schema or tablespace from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting a schema or directly enter a schema or tablespace. If the desired schema or tablespace is not displayed, check whether the login account has the permissions required to query metadata. **NOTE** The parameter value can contain wildcard characters (*), which is used to export all databases whose names start with a certain prefix or end with a certain suffix. For example: • **SCHEMA*** indicates that all databases whose names starting with **SCHEMA** are exported. • ***SCHEMA** indicates that all databases whose names ending with **SCHEMA** are exported. • ***SCHEMA*** indicates that all databases whose names containing **SCHEMA** are exported.	SCHEMA_E

Type	Parameter	Description	Example Value
	Table Name	Name of the table from which data will be extracted. This parameter is displayed when **Use SQL Statement** is set to **No**. Click the icon next to the text box to go to the page for selecting the table or directly enter a table name. If the desired table is not displayed, check whether the table exists or whether the login account has the permission to query metadata. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*). **NOTE** The table name can contain wildcard characters (*), which is used to export all tables whose names start with a certain prefix or end with a certain suffix. The number and types of fields in the tables must be the same. The examples are as follows: • **table*** indicates that all tables whose names starting with **table** are exported. • ***table** indicates that all tables whose names ending with **table** are exported. • ***table*** indicates that all tables whose names containing **table** are exported.	table

Type	Parameter	Description	Example Value
Advanced attributes	Partition Column	This parameter is displayed when **Use SQL Statement** is set to **No**, indicating that a field used to split data during data extraction. CDM splits a job into multiple tasks based on this field and executes the tasks concurrently. Fields with data distributed evenly are used, such as the sequential number field. Click the icon next to the text box to go to the page for selecting a field or directly enter a field. **NOTE** • The following types of partition columns are supported: CHAR, VARCHAR, LONGVARCHAR, TINYINT, SMALLINT, INTEGER, BIGINT, REAL, FLOAT, DOUBLE, NUMERIC, DECIMAL, BIT, BOOLEAN, DATE, TIME, and TIMESTAMP. It is recommended that the partition column have an index. • If the partition column type is CHAR, VARCHAR, or LONGVARCHAR, the column value cannot contain characters other than those in the ASCII character code table or Chinese characters.	id
	Where Clause	Where clause used to specify the data extraction range. This parameter is displayed when **Use SQL Statement** is set to **No**. If this parameter is not set, the entire table is extracted. You can set a date macro variable to extract data generated on a specific date. For details, see **Incremental Migration of Relational Databases**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	DS='${dateformat(yyyy-MM-dd,-1,DAY)}'
	Null in Partition Column	Whether the partition column can contain null values	Yes

4.6.4 Destination Job Parameters

4.6.4.1 To OBS

If the destination link of a job is an **OBS link**, that is, data is to be imported to OBS, configure the destination job parameters based on **Table 4-83**.

Advanced attributes are optional and not displayed by default. You can click **Show Advanced Attributes** to display them.

Table 4-83 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	Bucket Name	Name of the OBS bucket that data will be written to	bucket_2
	Write Directory	OBS directory to which data will be written. Do not add **/** in front of the directory name. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	directory/

Category	Parameter	Description	Example Value
	File Format	Format in which data is written. The options are as follows: • **CSV**: Data is written in CSV format, which is used for migrating data tables to files. • **Binary**: Files will be transferred directly. CDM writes the files without changing their format. This setting is suitable for file migration. If data is migrated between file-related data sources, such as FTP, SFTP, OBS, and HDFS, the value of **File Format** must the same as the source file format. **NOTE** • The format can only be CSV when the source link is an MRS Hive link. • If the source is an FTP/SFTP server, only the binary format is supported.	CSV
	Duplicate File Processing Method	This parameter is available when the migration source is HDFS. Files with the same name and size are identified as duplicate files. If there are duplicate files during data writing, the following methods are available: • Replace • Skip • Stop job For details, see **Incremental File Migration**.	Skip
Advanced attributes	Encryption	Whether to encrypt the uploaded data and the encryption mode. The options are as follows: • **None**: Data is written without encryption. • **KMS**: KMS in Data Encryption Workshop (DEW) is used for encryption. If KMS encryption is enabled, MD5 verification for data cannot be performed. For details, see **Encryption and Decryption During File Migration**.	KMS

Category	Parameter	Description	Example Value
	KMS ID	Data encryption key. This parameter is displayed when **Encryption** is set to **KMS**. Click next to the text box to select the KMS key that was created in DEW. • If the KMS key of the same project as that of the CDM cluster is used, you do not need to modify **Project ID**. • If the KMS key of another project is used, you need to modify **Project ID**.	53440ccb-3e73-4700-98b5-71ff5476e621
	Project ID	ID of the project to which KMS ID belongs. The default value is the ID of the project to which the current CDM cluster belongs. • If KMS and the CDM cluster are in the same project, retain the default value of **Project ID**. • If KMS of another project is used, set this parameter to the ID of the project to which KMS belongs.	9bd7c4bd54e5417198f9591bef07ae67
	Copy Content-Type	This parameter is displayed only when **File Format** is **Binary**, and both the migration source and destination are object storage. If you set this parameter to **Yes**, the Content-Type attribute of the source file is copied during object file migration. This function is mainly used for static website migration. The Content-Type attribute cannot be written to Archive buckets. Therefore, if you set this parameter to **Yes**, the migration destination must be a non-Archive bucket.	No
	Line Separator	Lind feed character in a file. By default, the system automatically identifies **\n**, **\r**, and **\r\n**. This parameter is not used when **File Format** is set to **Binary**.	\n
	Field Delimiter	Field delimiter in the file. This parameter is not used when **File Format** is set to **Binary**.	,
	File Size	This parameter is displayed only when the migration source is a database. Files are partitioned as multiple files by size so that they can be exported in proper size. The unit is MB.	1024

Category	Parameter	Description	Example Value
	Validate MD5 Value	The MD5 value can be verified only when files are transferred in **Binary** format. KMS encryption cannot be used if the MD5 value needs to be verified. Calculate the MD5 value of the source files and verify it with the MD5 value returned by OBS. If an MD5 file exists on the migration source, the system directly reads the MD5 file from the migration source and verifies it with the MD5 value returned by OBS. For details, see **MD5 Verification**.	Yes
	Record MD5 Verification Result	Whether to record the MD5 verification result when **Validate MD5 Value** is set to **Yes**	Yes
	Record MD5 Link	OBS link to which the MD5 verification result will be written	obslink
	Record MD5 Bucket	OBS bucket to which the MD5 verification result will be written	cdm05
	Record MD5 Directory	Directory to which the MD5 verification result will be written	/md5/
	Encoding Type	Encoding type, for example, **UTF-8** or **GBK**. This parameter is not used when **File Format** is set to **Binary**.	GBK
	Use Quote Character	This parameter is displayed only when **File Format** is **CSV**. It is used when database tables are migrated to file systems. If you set this parameter to **Yes** and a field in the source data table contains a field delimiter or line separator, CDM uses double quotation marks (") as the quote character to quote the field content as a whole to prevent a field delimiter from dividing a field into two fields, or a line separator from dividing a field into different lines. For example, if the **hello,world** field in the database is quoted, it will be exported to the CSV file as a whole.	No

Category	Parameter	Description	Example Value
	Use First Row as Header	This parameter is displayed only when data is exported from a relational database to OBS and **File Format** is set to **CSV**. When a table is migrated to a CSV file, CDM does not migrate the heading line of the table by default. If you set this parameter to **Yes**, CDM writes the heading line of the table to the file.	No
	Job Success Marker File	Whether to generate a marker file with a custom name in the destination directory after a job is executed successfully. If you do not specify a file name, this function is disabled by default.	finish.txt
	Folder Mode	This parameter is available only when data is exported from a relational database to OBS. If this function is enabled, generated files are named in the following format: *Root directory-Table name-Data type-Data folder format*. Example: **raw_schema/tbl_student/datas/tbl_student_1.csv**	Yes
	Blog/Clog File Name Extension	This parameter is available only when **Folder Mode** is set to **Yes**. It specifies the extension for the names of the files that contain custom Blob/Clog data in folder mode.	.dat/.jpg/.png
	Customize Hierarchical Directory	If this parameter is set to **Yes**, the files after migration can be stored in a custom directory. That is, only files are migrated. The directories to which the files belong are not migrated.	Yes
	Hierarchical Directory	Custom storage directory for files after migration. The time macro variable is supported. **NOTE** If the source link is a relational database link, the directory name consists of the source table name and a custom directory name. In other scenarios, the directory is a custom directory.	${dateformat(yyyy-MM-dd HH:mm:ss, -1, DAY)}

Category	Parameter	Description	Example Value
	Customize File Name	This parameter is displayed only when data is exported from a relational database to OBS and **File Format** is set to **CSV**. This parameter specifies the name of the file generated by OBS. The options are as follows: • **Character string**: Special characters are allowed. For example, if this parameter is set to **cdm#**, the name of the generated file is **cdm#.csv**. • Macro variable of time: If this parameter is set to **${timestamp()}**, the name of the generated file is **1554108737.csv**. • Macro variable of table name: If this parameter is set to **${tableName}**, the name of the generated file is the source table name **sqltabname.csv**. • Macro variable of version number: If this parameter is set to **${version}**, the name of the generated file is the cluster version number **2.9.2.200.csv**. • Any combination of the character string and macro variable (macro variable of time, table name, or version number). For example, if this parameter is set to **cdm#${timestamp()}_${version}**, the name of the generated file is **cdm#1554108737_2.9.2.200.csv**.	cdm

4.6.4.2 To HDFS

If the destination link of a job is an **HDFS link**, configure the destination job parameters based on **Table 4-84**.

Table 4-84 Parameter description

Parameter	Description	Example Value
Write Directory	HDFS directory to which data will be written. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	/user/output
File Format	Format in which data is written. The options are as follows: • **CSV**: Data is written in CSV format, which is used for migrating data tables to files. • **Binary**: Files will be transferred directly. CDM writes the files without changing their format. This setting is suitable for file migration. If data is migrated between file-related data sources, such as FTP, SFTP, OBS, and HDFS, the value of **File Format** must the same as the source file format.	CSV
Duplicate File Processing Method	This parameter is available when the migration source is a file data source, such as HTTP, FTP, SFTP, OBS, and HDFS. Files with the same name and size are identified as duplicate files. If there are duplicate files during data writing, the following methods are available: • Replace • Skip • Stop job	Stop job

Parameter	Description	Example Value
Compression Format	File compression format after data writing. The following compression formats are supported: • **None**: The files are not compressed. • **DEFLATE**: The files are compressed in DEFLATE format. • **gzip**: The files are compressed in gzip format. • **bzip2**: The files are compressed in bzip2 format. • **LZ4**: The files are compressed in LZ4 format. • **Snappy**: The files are compressed in snappy format.	Snappy
Line Separator	Lind feed character in a file. By default, the system automatically identifies **\n**, **\r**, and **\r\n**. This parameter is not used when **File Format** is set to **Binary**.	\n
Field Delimiter	Field delimiter in the file. This parameter is not used when **File Format** is set to **Binary**.	,
Use Quote Character	This parameter is displayed only when **File Format** is **CSV**. It is used when database tables are migrated to file systems. If you set this parameter to **Yes** and a field in the source data table contains a field delimiter or line separator, CDM uses double quotation marks (") as the quote character to quote the field content as a whole to prevent a field delimiter from dividing a field into two fields, or a line separator from dividing a field into different lines. For example, if the **hello,world** field in the database is quoted, it will be exported to the CSV file as a whole.	No
Use First Row as Header	When a table is migrated to a CSV file, CDM does not migrate the heading line of the table by default. If you set this parameter to **Yes**, CDM writes the heading line of the table to the file.	No
Write to Temporary File	Whether to write the binary file to a **.tmp** file first. After the migration is successful, run the **rename** or **move** command at the migration destination to restore the file.	No
Job Success Marker File	Whether to generate a marker file with a custom name in the destination directory after a job is executed successfully. If you do not specify a file name, this function is disabled by default.	finish.txt

Parameter	Description	Example Value
Customize Hierarchical Directory	Users can customize the directory hierarchy of files. Example: [Table name]/[Year]/[Month]/[Day]/[Data file name]. csv	-
Hierarchical Directory	Used to specify the directory level of a file, with time macro supported (the time format is yyyy/MM/dd). If this parameter is left blank, the directory does not have a hierarchical structure. **NOTE** If the source link is a relational database link, the directory name consists of the source table name and a custom directory name. In other scenarios, the directory is a custom directory.	${dateformat(yyyy/MM/dd, -1, DAY)}
Encryption	This parameter is displayed only when **File Format** is set to **Binary**. Whether to encrypt the uploaded data. The options are as follows: • **None**: Data is written without encryption. • **AES-256-GCM**: The AES 256-bit encryption algorithm is used to encrypt data. Currently, only the AES-256-GCM (NoPadding) encryption algorithm is supported. This parameter is used for encryption at the migration destination and decryption at the migration source. For details, see **Encryption and Decryption During File Migration**.	AES-256-GCM
DEK	This parameter is displayed only when **Encryption** is set to **AES-256-GCM**. The key consists of 64 hexadecimal numbers. Remember the key configured here because the decryption key must be the same as that configured here. If the encryption and decryption keys are inconsistent, the system does not report an exception, but the decrypted data is incorrect.	DD0AE00DFECD78BF051BCFDA25BD4E320DB0A7AC75A1F3FC3D3C56A457DCDC1B
IV	This parameter is displayed only when **Encryption** is set to **AES-256-GCM**. The initialization vector consists of 32 hexadecimal numbers. Remember the initialization vector configured here because the initialization vector used for decryption must be the same as that configured here. If the encryption and decryption keys are inconsistent, the system does not report an exception, but the decrypted data is incorrect.	5C91687BA886EDCD12ACBC3FF19A3C3F

NOTE

HDFS supports the **UTF-8** encoding only. Retain the default value **UTF-8**.

4.6.4.3 To HBase/CloudTable

If the destination link of a job is an **HBase link** or **CloudTable link**, configure the destination job parameters based on **Table 4-85**.

Table 4-85 Parameter description

Parameter	Description	Example Value
Table Name	Name of the HBase table to which data will be written. If you want to create an HBase table, you can copy the field names from the migration source. Click the icon next to the text box. The dialog box for selecting the table is displayed. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	TBL_2
Clear Data Before Import	Whether the data in the destination table is cleared before data import. The options are as follows: • **Yes**: The data is cleared. • **No**: The data is not cleared. Instead, it will be added to the existing table.	Yes

Parameter	Description	Example Value
Auto Table Creation	This parameter is displayed only when the source is a relational database. The options are as follows: ● **Non-auto creation**: CDM will not automatically create a table. ● **Auto creation**: If the destination database does not contain the table specified by **Table Name**, CDM will automatically create the table. If the table specified by **Table Name** already exists, no table is created and data is written to the existing table. **NOTE** The automatically created HBase table contains the column family and coprocessor information. For other attributes, default values are retained.	Non-auto creation
Rowkey Delimiter	(Optional) Used to combine multiple columns as a rowkey. Spaces are used by default.	,
Rowkey Data Redundancy	(Optional) Whether to write the rowkey data into HBase columns. The default value is **No**.	No
Compression Format	(Optional) Compression format used in creating an HBase table. The default value is **None**. ● **None**: The files are not compressed. ● **Snappy**: The files are compressed in snappy format. ● **gzip**: The files are compressed in gzip format.	None
Write WAL	Whether to enable Write Ahead Log (WAL) of HBase. The options are as follows: ● **Yes**: If the HBase server breaks down after the function is enabled, you can replay the operations that have not been performed in WAL. ● **No**: If you set this parameter to **No**, the write performance is improved. However, if the HBase server breaks down, data may be lost.	No

Parameter	Description	Example Value
Match Data Type	• **Yes**: Data of the Short, Int, Long, Float, Double, and Decimal columns in the source database is converted into Byte[] arrays (binary) and written into HBase. Other types of data are written as character strings. If several types of data mentioned above are combined as rowkeys, they will be written as character strings. This function saves storage space. In specific scenarios, the rowkey distribution is evener. • **No**: All types of data in the source database are written into HBase as character strings.	No

4.6.4.4 To Hive

If the destination link of a job is a **Hive link**, configure the destination job parameters based on **Table 4-86**.

Table 4-86 Parameter description

Parameter	Description	Example Value
Database Name	Database name. Click the icon next to the text box. The dialog box for selecting the database is displayed.	default
Table Name	Destination table name. Click the icon next to the text box. The dialog box for selecting the table is displayed. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	TBL_X

Parameter	Description	Example Value
Auto Table Creation	This parameter is displayed only when the source is a relational database. The options are as follows: • **Non-auto creation**: CDM will not automatically create a table. • **Auto creation**: If the destination database does not contain the table specified by **Table Name**, CDM will automatically create the table. If the table specified by **Table Name** already exists, no table is created and data is written to the existing table. • **Deletion before creation**: CDM deletes the table specified by **Table Name**, and then creates the table again. **NOTE** • Only column comments are synchronized during automatic table creation. Table comments are not synchronized. • Primary keys cannot be synchronized during automatic table creation.	Non-auto creation
Clear Data Before Import	Whether the data in the destination table is cleared before data import. The options are as follows: • **Yes**: The data is cleared. • **No**: The data is not cleared. Instead, it will be added to the existing table.	Yes
Partition to Clear	This parameter is available when **Clear Data Before Import** is set to **Yes**. When you enter the information about the partitions to be cleared, the data in the partitions will be cleared.	Single partition: **year=2020,location=sun** Multiple partitions: **['year=2020,location=sun', 'year=2021,location=earth']**

Parameter	Description	Example Value
Executing Analyze Statements	After all data is written, the ANALYZE TABLE statement is asynchronously executed to accelerate the Hive table query. The SQL statement is as follows: ● Non-partitioned table: **ANALYZE TABLE tablename COMPUTE STATISTICS** ● Partitioned table: **ANALYZE TABLE tablename PARTITION(partcol1[=val1], partcol2[=val2], ...) COMPUTE STATISTICS** **NOTE** Parameter **Executing Analyze Statements** applies only to the migration of a single table.	Yes

NOTE

- When Hive serves as the destination end, a table whose storage format is ORC is automatically created.
- Due to file format restrictions, complex data can be written only in ORC or Parquet format.
- If the source Hive contains both the array and map types of data, the destination table format can only be the ORC or parquet complex type. If the destination table format is RC or TEXT, the source data will be processed and can be successfully written.
- As the map type is an unordered data structure, the data type may change after a migration.
- If Hive serves as the migration destination and the storage format is Textfile, delimiters must be explicitly specified in the statement for creating Hive tables. The following is an example:

```
CREATE TABLE csv_tbl(
smallint_value smallint,
tinyint_value tinyint,
int_value int,
bigint_value bigint,
float_value float,
double_value double,
decimal_value decimal(9, 7),
timestmamp_value timestamp,
date_value date,
varchar_value varchar(100),
string_value string,
char_value char(20),
boolean_value boolean,
binary_value binary,
varchar_null varchar(100),
string_null string,
char_null char(20),
int_null int
)
ROW FORMAT SERDE 'org.apache.hadoop.hive.serde2.OpenCSVSerde'
WITH SERDEPROPERTIES (
"separatorChar" = "\t",
"quoteChar"     = "'",
"escapeChar"    = "\\"
)
STORED AS TEXTFILE;
```


4.6.4.5 To MySQL/SQL Server/PostgreSQL

Table 4-87 lists the destination job parameters when the destination link is an MySQL, SQL Server, or PostgreSQL link.

Table 4-87 Parameter description

Category	Parameter	Description	Example Value
Basic parameters	Schema/ Tablespace	Name of the database to which data will be written. The schema can be automatically created. Click the icon next to the text box to select a schema or tablespace.	schema
	Auto Table Creation	This parameter is displayed only when the source is a relational database. The options are as follows: • **Non-auto creation**: CDM will not automatically create a table. • **Auto creation**: If the destination database does not contain the table specified by **Table Name**, CDM will automatically create the table. If the table specified by **Table Name** already exists, no table is created and data is written to the existing table. • **Deletion before creation**: CDM deletes the table specified by **Table Name**, and then creates the table again.	Non-auto creation
	Table Name	Name of the table to which data will be written. Click the icon next to the text box. The dialog box for selecting the table is displayed. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	table

Category	Parameter	Description	Example Value
	Clear Data Before Import	Whether to clear the data in the destination table before data import. The options are as follows: • **Do not clear**: The data in the destination table is not cleared before data import. The imported data is just added to the table. • **Clear all data**: All data is cleared from the destination table before data import. • **Clear part of data**: Part of the data in the destination table is cleared before data import. If you select **Clear part of data**, you must configure **WHERE Clause** to specify which part will be deleted.	Clear part of data
	WHERE Clause	If **Clear Data Before Import** is set to **Clear part of data**, data in the destination table will be deleted based on the WHERE clause after the configuration is complete and before the import starts.	age > 18 and age <= 60
	Constraint Conflict Handling	How to handle data conflicts when data is being imported to RDS for MySQL • **insert into**: When a primary key or unique index conflict occurs, data cannot be written and will become dirty data. • **replace into**: When a primary key or unique index conflict occurs, the original row is deleted and a new row is inserted to replace all the fields in the original row. • **on duplicate key update**: When a primary key or unique index conflict occurs in a row in the destination table, the data columns except the unique constraint column in this row are updated.	insert into

Category	Parameter	Description	Example Value
Advanced parameters	Import to Staging Table	If you set this parameter to **Yes**, the transaction mode is enabled. CDM automatically creates a temporary table and imports data to the temporary table. After the data is imported successfully, it is migrated to the destination table in transaction mode. If the import fails, the destination table is rolled back to the state before the job starts. For details, see **Migration in Transaction Mode**. The default value is **No**, indicating that CDM directly imports the data to the destination table. In this case, if the job fails to be executed, the data that has been imported to the destination table will not be rolled back automatically. **NOTE** If you select **Clear part of data** or **Clear all data** for **Clear Data Before Import**, CDM does not roll back the deleted data in transaction mode.	No
	Extend Field Length	When **Auto creation** is selected, the length of the character fields can be extended to three times the original length and then written to the destination table. If the encoding types of the source and destination databases are different, but the character fields in the source and destination tables are the same, errors may occur during data migration due to character length difference. **NOTE** When this function is enabled, some fields consume three times the storage space of the user.	No
	Use NOT NULL Constraint	If you choose to create a target table automatically and specify the NOT NULL constraint, keep the NOT NULL constraints of the source and target tables consistent.	Yes
	Prepare for Data Import	The SQL statement that is first executed before a task is executed. Currently, only one SQL statement can be executed in wizard mode.	create temp table
	Complete Statement After Data Import	The SQL statement that is executed after a task is executed. Currently, only one SQL statement can be executed.	merge into

Category	Parameter	Description	Example Value
	Loader Threads	Number of threads started in each loader. A larger number allows more concurrent write operations. **NOTE** This parameter is unavailable if **Constraint Conflict Handling** is set to **replace into** or **on duplicate key update**.	1

4.6.4.6 To Oracle

If the destination link of a job is an **Oracle database link**, configure the destination job parameters based on **Table 4-88**.

Table 4-88 Parameter description

Type	Parameter	Description	Example Value
Basic parameters	Schema/ Tablespace	Name of the database to which data will be written. The schema can be automatically created. Click the icon next to the text box to select a schema or tablespace.	schema
	Table Name	Name of the table to which data will be written. Click the icon next to the text box. The dialog box for selecting the table is displayed. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	table

Type	Parameter	Description	Example Value
	Clear Data Before Import	Whether to clear the data in the destination table before data import. The options are as follows: • **Do not clear**: The data in the destination table is not cleared before data import. The imported data is just added to the table. • **Clear all data**: All data is cleared from the destination table before data import. • **Clear part of data**: Part of the data in the destination table is cleared before data import. If you select **Clear part of data**, you must configure **WHERE Clause** to specify which part will be deleted.	Clear part of data
	WHERE Clause	If **Clear Data Before Import** is set to **Clear part of data**, data in the destination table will be deleted based on the WHERE clause after the configuration is complete and before the import starts.	age > 18 and age <= 60
Advanced parameters	Import to Staging Table	If you set this parameter to **Yes**, the transaction mode is enabled. CDM automatically creates a temporary table and imports data to the temporary table. After the data is imported successfully, it is migrated to the destination table in transaction mode. If the import fails, the destination table is rolled back to the state before the job starts. For details, see **Migration in Transaction Mode**. The default value is **No**, indicating that CDM directly imports the data to the destination table. In this case, if the job fails to be executed, the data that has been imported to the destination table will not be rolled back automatically. **NOTE** If you select **Clear part of data** or **Clear all data** for **Clear Data Before Import**, CDM does not roll back the deleted data in transaction mode.	No
	Prepare for Data Import	The SQL statement that is first executed before a task is executed. Currently, only one SQL statement can be executed in wizard mode.	create temp table

Type	Parameter	Description	Example Value
	Complete Statement After Data Import	The SQL statement that is executed after a task is executed. Currently, only one SQL statement can be executed.	merge into
	Loader Threads	Number of threads started in each loader. A larger number allows more concurrent write operations. **NOTE** This parameter is unavailable if **Constraint Conflict Handling** is set to **replace into** or **on duplicate key update**.	1

4.6.4.7 To DWS

If the destination link of a job is a **DWS link**, configure the destination job parameters based on **Table 4-89**.

Table 4-89 Parameter description

Parameter	Description	Example Value
Schema / Tablespace	Name of the database to which data will be written. The schema can be automatically created. Click the icon next to the text box to select a schema or tablespace.	schema

Parameter	Description	Example Value
Auto Table Creation	This parameter is displayed only when the source is a relational database. The options are as follows: • **Non-auto creation**: CDM will not automatically create a table. • **Auto creation**: If the destination database does not contain the table specified by **Table Name**, CDM will automatically create the table. If the table specified by **Table Name** already exists, no table is created and data is written to the existing table. • **Deletion before creation**: CDM deletes the table specified by **Table Name**, and then creates the table again. **Field Mapping in Automatic Table Creation on DWS** describes the field mapping between the DWS tables created by CDM and source tables. **NOTE** Only column comments are synchronized during automatic table creation. Table comments are not synchronized.	Non-auto creation
Table Name	Name of the table to which data will be written. Click the icon next to the text box. The dialog box for selecting the table is displayed. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	table
Compress Data	Whether to compress data when data is imported to DWS and **Auto creation** is selected	No

Parameter	Description	Example Value
Storage Mode	When data is imported to DWS and **Auto Creation** is selected, you can specify the data storage mode: • **Row-based**: Row-based storage. It is used for point queries (index-based simple queries with fewer return records), or the scenario that requires a large number of addition, deletion, and modification operations. • **Column-based**: Column-based storage. It is used for statistical analysis queries (group and join scenarios) or ad hoc queries (query conditions are uncertain and indexes can hardly be used to scan row-based tables).	Row-based
Import Mode	Mode for importing data to DWS • In COPY mode, the source data is copied to the DataNode of DWS after passing through the management node. • In UPSERT mode, if a primary key or unique constraint conflict occurs, other data columns, except the primary key and unique constraint column, are updated.	COPY
Clear Data Before Import	Whether to clear the data in the destination table before data import. The options are as follows: • **Do not clear**: The data in the destination table is not cleared before data import. The imported data is just added to the table. • **Clear all data**: All data is cleared from the destination table before data import. • **Clear part of data**: Part of the data in the destination table is cleared before data import. If you select **Clear part of data**, you must configure **WHERE Clause** to specify which part will be deleted.	Clear part of data
WHERE Clause	If **Clear Data Before Import** is set to **Clear part of data**, data in the destination table will be deleted based on the WHERE clause after the configuration is complete and before the import starts.	age > 18 and age <= 60

Parameter	Description	Example Value
Import to Staging Table	If you set this parameter to **Yes**, the transaction mode is enabled. CDM automatically creates a temporary table and imports data to the temporary table. After the data is imported successfully, it is migrated to the destination table in transaction mode. If the import fails, the destination table is rolled back to the state before the job starts. . The default value is **No**, indicating that CDM directly imports the data to the destination table. In this case, if the job fails to be executed, the data that has been imported to the destination table will not be rolled back automatically. **NOTE** If you select **Clear part of data** or **Clear all data** for **Clear Data Before Import**, CDM does not roll back the deleted data in transaction mode.	No
Extending field length	When **Auto creation** is selected, the length of the character fields can be extended to three times the original length and then written to the destination table. If the encoding types of the source and destination databases are different, but the character fields in the source and destination tables are the same, errors may occur during data migration due to character length difference. When a character field containing Chinese characters is imported to DWS, the length of the character field must be automatically increased by three times. If a job fails to be executed and an error message similar to **value too long for type character varying** exists in the log when you import Chinese characters to DWS, you can enable this function to solve the problem. **NOTE** When this function is enabled, some fields consume three times the storage space of the user.	No
Use NOT NULL Constraint	If you choose to create a target table automatically and specify the NOT NULL constraint, keep the NOT NULL constraints of the source and target tables consistent.	Yes
Prepare for Data Import	The SQL statement that is first executed before a task is executed. Currently, only one SQL statement can be executed in wizard mode.	create temp table

Parameter	Description	Example Value
Complete Statement After Data Import	The SQL statement that is executed after a task is executed. Currently, only one SQL statement can be executed.	merge into
Loader Threads	Number of threads started in each loader. A larger number allows more concurrent write operations.	1

Field Mapping in Automatic Table Creation on DWS

Figure 4-38 describes the field mapping between DWS tables created by CDM and source tables. For example, if you use CDM to migrate the Oracle database to DWS, CDM automatically creates a table on DWS and maps the **NUMBER(3,0)** field of the Oracle database to the **SMALLINT** field of DWS.

Figure 4-38 Field mapping in automatic table creation

Source Database Type							Destination Database Type
Oracle	MySQL	SQL Server	PostgreSQL	Db2	GaussDB	SAP HANA	DWS
NUMBER(p,0) (p=3 or p=5)	SMALLINT,TINYINT	SMALLINT,TINYINT	SMALLINT	DECIMAL	SMALLINT	SMALLINT,TINYINT	SMALLINT
NUMBER(10,0)	INT	INT	INTEGER	INT	INTEGER	INTEGER	INTEGER
NUMBER(19,0)	BIGINT	BIGINT	BIGINT	DECIMAL	BIGINT	BIGINT	BIGINT
None	None	None	OID	None	OID	CHAR(128)	OID
NUMBER(p,s) (0 < p <= 38)	DECIMAL(p,s) (0 < p <= 65)	DECIMAL(p,s) (0 < p <= 30)	NUMERIC(p,s) (p <= 1000)	DECIMAL	NUMERIC(p,s) (p <= 1000)	DECIMAL(p,s) (0 < p <= 38)	NUMERIC(p,s) (p <= 1000)
RAW	BINARY	BINARY	BYTEA	BINARY	BYTEA	BINARY	BYTEA
CHAR	CHAR	CHAR	CHAR	CHAR	CHAR	CHAR(p) (p <= 2000)	CHAR
NCHAR	NCHAR	NCHAR	NCHAR	NCHAR	NCHAR	NCHAR(p) (p <= 5000)	NCHAR
DATE	DATE	DATE	DATE	DATE	DATE	DATE	DATE
DATE	DATETIME	DATETIME2	TIMESTAMP	TIMESTAMP	TIMESTAMP	TIMESTAMP	TIMESTAMP
VARCHAR2(p) (p <= 4000)	VARCHAR	VARCHAR(p) (if p >= 8000 p=max)	VARCHAR(p) (p <= 10485760)	VARCHAR	VARCHAR(p) (p <= 10485760)	VARCHAR(p) (p <= 5000)	VARCHAR(p) (p <= 10485760)
FLOAT	DOUBLE	FLOAT	DOUBLE PRECISION	FLOAT	DOUBLE PRECISION	DOUBLE	DOUBLE PRECISION
FLOAT	REAL	FLOAT	REAL	FLOAT	REAL	REAL	REAL
CLOB	TEXT	TEXT	TEXT	TEXT	TEXT	CLOB	TEXT
DATE	None	TIME	TIME	TIME	TIME	TIME	TIME
BOOLEAN	None	None	BOOLEAN	BOOLEAN	BOOLEAN	BOOLEAN	BOOLEAN

NOTE

Indexes cannot be created in automatic table creation scenarios.

4.6.4.8 To DDS

If the destination link of a job is a **DDS link**, configure the destination job parameters based on **Table 4-90**.

Table 4-90 Parameter description

Parameter	Description	Example Value
Database Name	Database to which data is to be imported	ddsdb
Collection Name	Collection of data to be imported, which is similar to the table name of a relational database. Click the icon next to the text box to go to the page for selecting the table or directly enter a table name. If the desired table is not displayed, check whether the table exists or whether the login account has the permission to query metadata.	COLLECTION

4.6.4.9 To Redis

Table 4-91 lists the destination job parameters when the destination link is a Redis link.

Table 4-91 Parameter description

Parameter	Description	Example Value
Redis Key Prefix	Key prefix, which is similar to the table name of a relational database	TABLE
Value Storage Type	The options are as follows: • **String**: without column name, such as **value1,value2** • **Hash**: with column name, such as **column1=value1,column2=value2**	String
Use Column Value as Field	This parameter is displayed when **Value Storage Type** is set to **HASH**. Only **Hash** is supported. If this function is enabled, values are alternately used as fields and values in sequence except the primary key column.	Yes

Parameter	Description	Example Value
Delete Same Key Before Writing	Whether to delete the same key before writing. • **No**: If a key with the same name but of a different type already exists in Redis, the migration job skips the key. • **Yes**: Redis deletes the existing key with the same name and then performs the migration.	No
Key Delimiter	Character used to separate table names and column names of a relational database	_
Value Delimiter	Character used to separate columns when the storage type is string	;
Validity period of the key value	Unified time to live (TTL) of a key, in seconds	300

4.6.4.10 To Elasticsearch/CSS

If the destination link of a job is a link described in **Link to Elasticsearch** or **Link to CSS**, configure the destination job parameters based on **Table 4-92**.

NOTICE

The parameters required for table/file migration are different from those for entire DB migration. The following table lists the parameters for table/file migration. The actual parameters are subject to those displayed on the console.

Table 4-92 Job parameters when Elasticsearch/CSS is the destination

Parameter	Description	Example Value
Index	Elasticsearch index, which is similar to the name of a relational database. CDM supports automatic creation of indexes and field types. The index and field type names can contain only lowercase letters.	index

Parameter	Description	Example Value
Type	Elasticsearch type, which is similar to the table name of a relational database. The type name can contain only lowercase letters. **NOTE** Elasticsearch 7.x and later versions do not support custom types. Instead, only the **_doc** type can be used. In this case, this parameter does not take effect even if it is set.	type
Pipeline ID	ID of the pipeline used to convert the format of the data transferred to Elasticsearch. If the destination is Elasticsearch, you need to create a pipeline ID in Kibana first. If the destination is CSS, you do not need to create a pipeline ID. Instead, enter the name of the configuration file, which is **name** by default.	If the destination is Elasticsearch: pipeline_id If the destination is CSS: **name** (name of the configuration file)
Write ES with Routing	If you enable this function, a column can be written to Elasticsearch as a route. **NOTE** Before enabling this function, create indexes at the destination to improve the query efficiency.	No
Route Column	This parameter is available when **Write ES with Routing** is set to **Yes**. It specifies the destination routing column. If the destination index exists but the column information cannot be obtained, you can manually enter the column. The route column can be empty. If it is empty, no routing value is specified for the data written to Elasticsearch.	value1

Parameter	Description	Example Value
Periodically Create Index	For streaming jobs that continuously write data to Elasticsearch, CDM periodically creates indexes and writes data to the indexes, which helps you delete expired data. The indexes can be created based on the following periods: • **Every hour**: CDM creates indexes on the hour. The new indexes are named in the format of *Index name+Year+Month+Day+Hour*, for example, **index2018121709**. • **Every day**: CDM creates indexes at 00:00 every day. The new indexes are named in the format of *Index name+Year+Month+Day*, for example, **index20181217**. • **Every week**: CDM creates indexes at 00:00 every Monday. The new indexes are named in the format of *Index name+Year+Week*, for example, **index201842**. • **Every month**: CDM creates indexes at 00:00 on the first day of each month. The new indexes are named in the format of *Index name+Year +Month*, for example, **index201812**. • **Do not create**: Do not create indexes periodically. When extracting data from a file, you must configure a single extractor, which means setting **Concurrent Extractors** to **1**. Otherwise, this parameter is invalid.	Every hour

4.6.4.11 To DLI

If the destination link of a job is a **DLI link**, configure the destination job parameters based on **Table 4-93**.

NOTE

When you use CDM to migrate data to DLI, DLI generates data files in the *dli-trans** temporary OBS bucket. Therefore, you need to grant the account corresponding to the AK/SK the permissions to read and write the *dli-trans** bucket and create directories. For details about how to add OBS permission policies, see **Adding an OBS Bucket Policy**.

Table 4-93 Parameter description

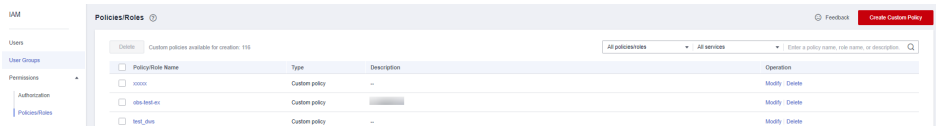
Parameter	Description	Example Value
Resource Queue	Resource queue to which the destination table belongs The default queue of DLI cannot be used for migration jobs. You need to create a SQL queue in DLI. For details about how to create a queue, see **Creating a Queue**.	cdm
Database Name	Name of the database to which data will be written	dli
Table Name	Name of the table to which data will be written	car_detail
Clear Data Before Import	Whether to clear data in the destination table before data import If this parameter is set to **Yes**, data in the destination table will be cleared before the task is started.	No
Convert empty strings to null	If this parameter is set to **Yes**, an empty string is regarded as null.	No
Data Clearing Mode	This parameter is available when **Clear Data Before Import** is set to **Yes**. **TRUNCATE**: deletes standard data. **INSERT_OVERWRITE**: overwrites existing data with inserted data. **NOTE** If the source link is a Kafka link and **Clear Data Before Import** is set to **Yes**, **INSERT_OVERWRITE** is unavailable.	TRUNCATE
Partition	This parameter is available when **Clear Data Before Import** is set to **Yes**. When you enter partitions, data in these partitions will be cleared.	year=2020,location=sun

Adding an OBS Bucket Policy

Step 1 Log in to the IAM console.

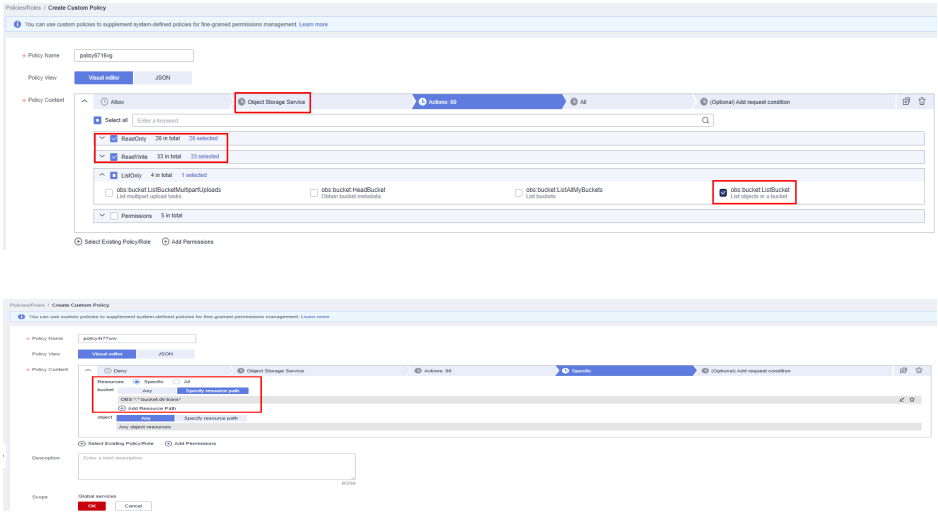
Step 2 In the navigation pane, choose **Permissions** > **Policies/Roles** and click **Create Custom Policy** in the upper right corner.

Figure 4-39 Creating a custom policy



Step 3 Enter a policy name and set **Policy Content**.

Figure 4-40 Configuring the policy



Step 4 Enter the policy description and click **OK**.

----End

4.6.4.12 To OpenTSDB

If the destination link of a job is a **CloudTable OpenTSDB link**, configure the destination job parameters based on **Table 4-94**.

Table 4-94 Parameter description

Parameter	Description	Example Value
Metric	(Optional) You can specify a metric or select an existing metric in OpenTSDB.	city.temp
Time	(Optional) Data point. The value is a string or timestamp in the format of *yyyyMMddHHmmdd*.	1598870800
Tag	(Optional) Data tag	tagk:tagv, tagk2:tagv2

4.6.4.13 To MRS Hudi

If the destination link of a job is an **MRS Hudi link**, configure the destination job parameters based on **Table 4-95**.

Table 4-95 Parameter description

General Configuration		
Item	Configuration Description	Recommended Configuration
Destination Link Name	MRS Hudi link	hudi_to_cdm
Database Name	Database name. Click the icon next to the text box. The dialog box for selecting the database is displayed.	dbadmin
Table Name	Click the icon next to the text box. The dialog box for selecting the table is displayed. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. You can use macro variables of date and time in a scheduled job to synchronize incremental data periodically. For details, see **Using Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	cdm
Auto Table Creation	Whether to automatically create Hudi tables • **Non-auto creation**: CDM will not automatically create a table. • **Auto creation**: If the destination database does not contain the table specified by **Table Name**, CDM will automatically create the table. If the table specified by **Table Name** already exists, no table is created and data is written to the existing table.	Non-auto creation
Clear Data Before Import	Whether the data in the destination table is cleared before data import. The options are as follows: • **Yes**: The data is cleared. • **No**: The data is not cleared. Instead, it will be added to the existing table.	No

General Configuration		
Full Data Mode to Write Hoodie	Hoodie write mode. The default value is **Yes**, indicating the full mode. Value **No** indicates the microbatch mode. • In full mode, data is asynchronously written to Hoodie by fragments, which is suitable for writing all data at a time. • In microbatch mode, data is asynchronously written to Hoodie in batches. This mode is suitable if there are strict SLA requirements on the import time, a small number of resources are required, or the MOR table storage types are compressed online. **NOTE** This mode cannot be changed during a retry upon failure.	Yes
Batch Size	This parameter is available when **Full Data Mode to Write Hoodie** is set to **No**. It specifies the number of data rows written to Hoodie in a single batch. The default value is **100000**.	100000
Use the import time field	A field marked as the import time field. If a table is automatically created, this field is automatically added to the table creation statement. When data is written to Hudi, the value of this field is replaced by the current time. If the table is not automatically created, select the existing import time field.	Yes
Data import time field name	This parameter is available when **Use the import time field** is set to **Yes**. It specifies the time when data is written to Hudi. **NOTE** • If the destination table already has an import time field, you can directly use the existing timestamp field. • In the automatic table creation scenario, this field is concatenated to the table creation statement and it is a timestamp. The field name cannot be the same as that of any source field (including custom fields).	cdc_last_update_date
Hudi Table Creation Configuration		

General Configuration		
Location	OBS or HDFS path where database table files are stored	-
Hudi Table Type	Storage type of the Hudi table • **MOR**: Data is written to a log file in avro format and then merged into a Parquet file when being read. • **COW**: Data is directly written to a Parquet file.	MOR
Hudi table primary key	Primary keys for creating a Hudi table. Use commas (,) to separate multiple keys.	-
Hudi Table Key Generator Class	Primary key generation type, which implements **org.apache.hudi.keygen.KeyGenerator** to extract key values from input records.	-
Hudi table pre-combine key	If two records have the same primary key, the record with a larger **precombine** value is retained. **NOTE** If no time field is available, you can set a field that is the same as the primary key. When a primary key conflict occurs, the latest record is retained.	ts
Hudi Table Partition Fields	Partition fields for creating a Hudi table. Use commas (,) to separate multiple fields.	-
Hudi table compression policy (whether to enable write compression)	Policy for compressing data online. This parameter takes effect only for MOR tables.	Yes
Hudi Table Clean Policy (Reserved Submissions)	Number of submissions reserved during clearance	1
Hudi Table Archiving Policy (Minimum Retention Submissions)	Minimum number of submissions retained during archiving	1

General Configuration		
Hudi Table Archiving Policy (Maximum Number of Retained Submissions)	Maximum number of submissions retained during archiving	100
Hudi table options	Custom parameters for creating a Hudi table. The parameters take effect in options, for example, **primary key**, **combineKey**, or **index**.	-

4.6.4.14 To MRS ClickHouse

If the destination link of a job is an **MRS ClickHouse link**, configure the destination job parameters based on **Table 4-96**.

NOTE

If the source link of the job is an MRS ClickHouse, DWS, or Hive link:

- If the int or float fields are null, set the field type to **nullable()** when creating an MRS ClickHouse table. Otherwise, the value written to MRS ClickHouse is **0**.
- Check whether the destination table engine is ReplicatedMergeTree. This engine has a deduplication mechanism, in which the data to be deduplicated cannot be predicted accurately. If this engine is used, ensure that data is unique. Otherwise, non-unique data will be ignored and not written, or ReplicatedMergeTree will be replaced by other types of table engines such as MergeTree.

Table 4-96 Parameter description

Parameter	Description	Example Value
Schema/ Tablespace	Click the icon next to the text box to select a schema or tablespace.	schema

Parameter	Description	Example Value
Table Name	Destination table name. Click the icon next to the text box. The dialog box for selecting the table is displayed. This parameter can be configured as a macro variable of date and time and a path name can contain multiple macro variables. When the macro variable of date and time works with a scheduled job, the incremental data can be synchronized periodically. For details, see **Incremental Synchronization Using the Macro Variables of Date and Time**. **NOTE** If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).	table
Clear Data Before Import	Whether to clear the data in the destination table before data import. The options are as follows: • **Do not clear**: The data in the destination table is not cleared before data import. The imported data is just added to the table. • **Clear all data**: All data is cleared from the destination table before data import. • **Clear part of data**: Part of the data in the destination table is cleared before data import. If you select **Clear part of data**, you must configure **WHERE Clause** to specify which part will be deleted.	Clear part of data
Whether On Cluster	This parameter is displayed when **Clear Data Before Import** is set to **Clear part of data** or **Clear all data**. If this parameter is set to **Yes**, all or part of data on all the nodes in the cluster will be cleared.	Yes
WHERE Clause	If **Clear Data Before Import** is set to **Clear part of data**, data in the destination table will be deleted based on the WHERE clause after the configuration is complete and before the import starts.	age > 18 and age <= 60

4.6.4.15 To MongoDB

If the destination link of a job is a **MongoDB link**, configure the destination job parameters based on **Table 4-97**.

Table 4-97 Parameter description

Parameter	Description	Example Value
Database Name	Database to which data is to be imported	mddb
Collection Name	Collection of data to be imported, which is similar to the table name of a relational database. Click the icon next to the text box to go to the page for selecting the table or directly enter a table name. If the desired table is not displayed, check whether the table exists or whether the login account has the permission to query metadata.	COLLECTION
Behavior	Insert operation to be performed during record migration to the MongoDB ● **Insert**: Insert file records into a specified set. ● **Insert**: Use a specified filter key as the query condition. If a matching record is found in the set, the record is replaced. (If multiple matching records are found, only the first found record is replaced.) Otherwise, the new record will be added. ● **Replace**: Use a specified filter key as the query condition. If a matching record is found in the set, the record is replaced. (If multiple matching records are found, only the first found record is replaced.) Otherwise, the new record will not be added.	Add
Prepare for Data Import	MongoDB query statement that needs to be executed before a task is executed **NOTE** ● The value is a JSON string that contains two key-value pairs. The first key-value pair specifies the operation type. The key is **type**, and the value can only be **remove** or **drop**. The second key-value pair is the name of the data condition or set to be configured for the operation type. ● The execution of the data import preparation statement does not affect the data to be written.	{"type":"remove","json":"{$or:[{Pid:{$gt:'0',$lt:'2'}},{X:{$gt:'50',$lt:'80'}}]}"}

4.6.5 Configuring Field Mapping

Scenario

- After the job parameters are configured, you can configure field mapping. You can click on the **Map Field** page to customize new fields or click in the **Operation** column to create a field converter.

- If files are migrated between FTP, SFTP, OBS, and HDFS and the migration source's **File Format** is set to **Binary**, files will be directly transferred, free from field mapping.
- In other scenarios, CDM automatically maps fields of the source table and the destination table. You need to check whether the mapping and time format are correct. For example, check whether the source field type can be converted into the destination field type.
- In the auto table creation scenario, you need to add fields to the destination table in advance, and add the fields to the field mapping..

Constraints

- If **Use SQL Statement** is set to **Yes** in the source job configuration, converters cannot be created.
- On the **Map Field** tab page, if CDM fails to obtain all columns by obtaining sample values (for example, when data is exported from HBase, CloudTable, or MongoDB, there is a high probability that CDM failed to obtain all columns), you can click  and select **Add a new field** to add new fields to ensure that the data imported to the migration destination is complete.
- When a relational database, Hive, DLI, or MRS Hudi is used as the migration source, sample values cannot be obtained.
- When SQLServer is the destination, fields of the timestamp type cannot be written. You must change their type (for example, to datetime) so that they can be written.
- Column names are displayed when the source of the migration job is OBS, CSV files are to be migrated, and parameter **Extract first row as columns** is set to **Yes**.
- Field mapping is not involved when the binary format is used to migrate files to files.
- In the automatic table creation scenario, you need to manually add fields to the destination table in advance and then add fields to the field mapping.
- After a field is added, its sample value is not displayed on the console. This does not affect the field value transmission. CDM directly writes the field value to the destination end.
- If the field mapping is incorrect, you can adjust the field mapping by dragging fields or clicking  to map fields in batches.
- If the data is imported to DWS, you need to select the distribution columns in the destination fields. You are advised to select the distribution columns according to the following principles:

  a. Use the primary key as the distribution column.

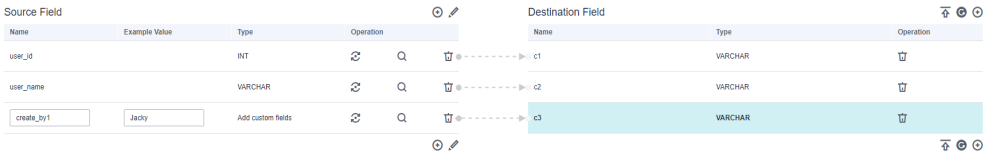
  b. If multiple data segments are combined as primary keys, specify all primary keys as the distribution column.

  c. In the scenario where no primary key is available, if no distribution column is selected, DWS uses the first column as the distribution column by default. As a result, data skew risks exist.
- If a source field type is not supported, convert the field type to a type supported by CDM by referring to **Converting Unsupported Data Types**.

Adding a Field

You can click on the **Map Field** page and select **Add** to customize a new field. This field is usually used to mark the database source to ensure the integrity of the data imported to the migration destination.

Figure 4-41 Field mapping



Currently, the following field types are supported:

- **Constant Parameter**

  Constant parameters are fixed parameters and do not need to be reconfigured. For example, **lable** = **friends** is used to identify a constant value.

- **Variables**

  You can use variables such as time macros, table name macros, and version macros to mark database source information. The variable syntax is ${variable}, where **variable** indicates a variable. For example, **input_time** = **${timestamp()}** indicates the timestamp of the current time.

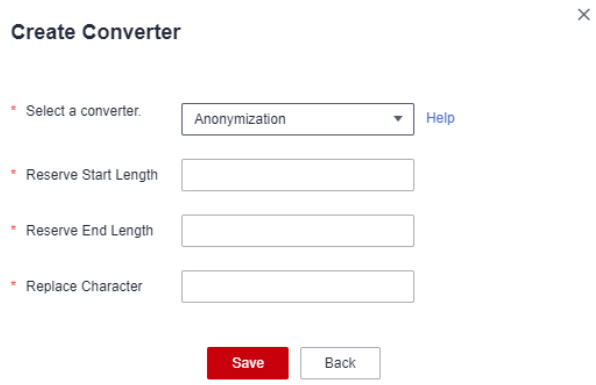
- Expression

  You can use the expression language to dynamically generate parameter values based on the running environment. The expression syntax is #{expr}, where **expr** indicates an expression. For example, **time** = **#{DateUtil.now()}** is used to identify the current date string.

Creating a Converter

CDM supports field conversion. Click and then click **Create Converter**.

Figure 4-42 Creating a converter



CDM can convert fields during migration. Currently, the following field converters are supported:

- **Anonymization**

  This converter is used to hide key information about the character string. For example, if you want to convert **12345678910** to **123****8910**, configure the parameters as follows:

  - Set **Reserve Start Length** to **3**.
  - Set **Reserve End Length** to **4**.
  - Set **Replace Character** to *****.

- **Trim**

  This converter is used to automatically delete the spaces before and after a string. No parameters need to be configured.

- **Reverse string**

  This converter is used to automatically reverse a string. For example, reverse **ABC** into **CBA**. No parameters need to be configured.

- **Replace string**

  This converter is used to replace a character string. You need to configure the object to be replaced and the new value.

- **Remove line break**

  This converter is used to delete the newline characters, such as \n, \r, and \r\n from the field.

- **Expression conversion**

  During data conversion, if the content to be replaced contains a special character, use a backslash (\) to escape the special character to a common one.

  - The expression supports the following environment variables:
    - **value**: indicates the current field value.
    - **row**: indicates the current row, which is an array type.
  - The expression supports the following Utils:
    i. If the field is of the string type, convert all character strings into lowercase letters, for example, convert **aBC** to **abc**.

       Expression: StringUtils.lowerCase(value)

    ii. Convert all character strings of the current field to uppercase letters.

       Expression: StringUtils.upperCase(value)

    iii. Convert the format of the first date field from 2018-01-05 15:15:05 to 20180105.

       Expression: DateUtils.format(DateUtils.parseDate(row[0],"yyyy-MM-dd HH:mm:ss"),"yyyyMMdd")

    iv. Convert a timestamp to a date string in *yyyy-MM-dd hh:mm:ss* format, for example, convert **1701312046588** to **2023-11-30 10:40:46**.

       Expression: DateUtils.format(NumberUtils.toLong(value),"yyyy-MM-dd HH:mm:ss")

    v. Convert a date string in the yyyy-MM-dd hh:mm:ss format to a timestamp.

Expression: DateUtils.getTime(DateUtils.parseDate(value,"yyyy-MM-dd hh:mm:ss"))

vi. If the field value is a date string in *yyyy-MM-dd* format, extract the year from the field value, for example, extract **2017** from **2017-12-01**.

Expression: StringUtils.substringBefore(value,"-")

vii. If the field value is of the numeric type, convert the value to a new value which is two times greater than the original value:

Expression: value*2

viii. Convert the field value **true** to **Y** and other field values to **N**.

Expression: value=="true"?"Y":"N"

ix. If the field value is of the string type and is left empty, convert it to **Default**. Otherwise, the field value will not be converted.

Expression: empty value? "Default":value

x. Convert date format **2018/01/05 15:15:05** to **2018-01-05 15:15:05**:

Expression: DateUtils.format(DateUtils.parseDate(value,"yyyy/MM/dd HH:mm:ss"),"yyyy-MM-dd HH:mm:ss")

xi. Obtain a 36-bit universally unique identifier (UUID):

Expression: CommonUtils.randomUUID()

xii. If the field is of the string type, capitalize the first letter, for example, convert **cat** to **Cat**.

Expression: StringUtils.capitalize(value)

xiii. If the field is of the string type, convert the first letter to a lowercase letter, for example, convert **Cat** to **cat**.

Expression: StringUtils.uncapitalize(value)

xiv. If the field is of the string type, use a space to fill in the character string to the specified length and center the character string. If the length of the character string is not shorter than the specified length, do not convert the character string. For example, convert **ab** to meet the specified length 4.

Expression: StringUtils.center(value,*4*)

xv. Delete a newline (including **\n**, **\r**, and **\r\n**) at the end of a character string. For example, convert **abc\r\n\r\n** to **abc\r\n**.

Expression: StringUtils.chomp(value)

xvi. If the string contains the specified string, **true** is returned; otherwise, **false** is returned. For example, **abc** contains **a** so that **true** is returned.

Expression: StringUtils.contains(value,"*a*")

xvii. If the string contains any character of the specified string, **true** is returned; otherwise, **false** is returned. For example, **zzabyycdxx** contains either **z** or **a** so that **true** is returned.

Expression: StringUtils.containsAny(value,"*za*")

xviii. If the string does not contain any one of the specified characters, **true** is returned. If any specified character is contained, **false** is returned. For example, **abz** contains one character of **xyz** so that **false** is returned.

Expression: StringUtils.containsNone(value,"*xyz*")

xix. If the string contains only the specified characters, **true** is returned. If any other character is contained, **false** is returned. For example, **abab** contains only characters among **abc** so that **true** is returned.

Expression: StringUtils.containsOnly(value,"*abc*")

xx. If the character string is empty or null, convert it to the specified character string. Otherwise, do not convert the character string. For example, convert the empty character string to null.

Expression: StringUtils.defaultIfEmpty(value,*null*)

xxi. If the string ends with the specified suffix (case sensitive), **true** is returned; otherwise, **false** is returned. For example, if the suffix of **abcdef** is not null, **false** is returned.

Expression: StringUtils.endsWith(value,*null*)

xxii. If the string is the same as the specified string (case sensitive), **true** is returned; otherwise, **false** is returned. For example, after strings **abc** and **ABC** are compared, **false** is returned.

Expression: StringUtils.equals(value,"*ABC*")

xxiii. Obtain the first index of the specified character string in a character string. If no index is found, **-1** is returned. For example, the first index of **ab** in **aabaabaa** is 1.

Expression: StringUtils.indexOf(value,"*ab*")

xxiv. Obtain the last index of the specified character string in a character string. If no index is found, **-1** is returned. For example, the last index of **k** in **aFkyk** is 4.

Expression: StringUtils.lastIndexOf(value,"*k*")

xxv. Obtain the first index of the specified character string from the position specified in the character string. If no index is found, **-1** is returned. For example, the first index of **b** obtained after the index 3 of **aabaabaa** is 5.

Expression: StringUtils.indexOf(value,"*b*",*3*)

xxvi. Obtain the first index of any specified character in a character string. If no index is found, **-1** is returned. For example, the first index of **z** or **a** in **zzabyycdxx.** is 0.

Expression: StringUtils.indexOfAny(value,"*za*")

xxvii. If the string contains any Unicode character, **true** is returned; otherwise, **false** is returned. For example, **ab2c** contains only non-Unicode characters so that **false** is returned.

Expression: StringUtils.isAlpha(value)

xxviii. If the string contains only Unicode characters and digits, **true** is returned; otherwise, **false** is returned. For example, **ab2c** contains only Unicode characters and digits, so that **true** is returned.

Expression: StringUtils.isAlphanumeric(value)

xxix. If the string contains only Unicode characters, digits, and spaces, **true** is returned; otherwise, **false** is returned. For example, **ab2c** contains only Unicode characters and digits, so that **true** is returned.

Expression: StringUtils.isAlphanumericSpace(value)

xxx. If the string contains only Unicode characters and spaces, **true** is returned; otherwise, **false** is returned. For example, **ab2c** contains Unicode characters and digits so that **false** is returned.

Expression: StringUtils.isAlphaSpace(value)

xxxi. If the string contains only printable ASCII characters, **true** is returned; otherwise, **false** is returned. For example, for **!ab-c~**, **true** is returned.

Expression: StringUtils.isAsciiPrintable(value)

xxxii. If the string is empty or null, **true** is returned; otherwise, **false** is returned.

Expression: StringUtils.isEmpty(value)

xxxiii. If the string contains only Unicode digits, **true** is returned; otherwise, **false** is returned.

Expression: StringUtils.isNumeric(value)

xxxiv. Obtain the leftmost characters of the specified length. For example, obtain the leftmost two characters **ab** from **abc**.

Expression: StringUtils.left(value,*2*)

xxxv. Obtain the rightmost characters of the specified length. For example, obtain the rightmost two characters **bc** from **abc**.

Expression: StringUtils.right(value,*2*)

xxxvi. Concatenate the specified character string to the left of the current character string and specify the length of the concatenated character string. If the length of the current character string is not shorter than the specified length, the character string will not be converted. For example, if **yz** is concatenated to the left of **bat** and the length must be 8 after concatenation, the character string is **yzyzybat** after conversion.

Expression: StringUtils.leftPad(value,*8*,"*yz*")

xxxvii. Concatenate the specified character string to the right of the current character string and specify the length of the concatenated character string. If the length of the current character string is not shorter than the specified length, the character string will not be converted. For example, if **yz** is concatenated to the right of **bat** and the length must be 8 after concatenation, the character string is **batyzyzy** after conversion.

Expression: StringUtils.rightPad(value,*8*,"*yz*")

xxxviii. If the field is of the string type, obtain the length of the current character string. If the character string is null, **0** is returned.

Expression: StringUtils.length(value)

xxxix. If the field is of the string type, delete all the specified character strings from it. For example, delete **ue** from **queued** to obtain **qd**.

Expression: StringUtils.remove(value,"*ue*")

xl. If the field is of the string type, remove the substring at the end of the field. If the specified substring is not at the end of the field, no conversion is performed. For example, remove **.com** at the end of **www.domain.com**.

Expression: StringUtils.removeEnd(value,"*.com*")

xli. If the field is of the string type, delete the substring at the beginning of the field. If the specified substring is not at the beginning of the field, no conversion is performed. For example, delete **www.** at the beginning of **www.domain.com**.

Expression: StringUtils.removeStart(value,"*www.*")

xlii. If the field is of the string type, replace all the specified character strings in the field. For example, replace **a** in **aba** with **z** to obtain **zbz**.

Expression: StringUtils.replace(value,"*a*","*z*")

If the content to be replaced contains a special character, the special character must be escaped to a common character. For example, if you want to delete **\t** from a string, use the following expression: StringUtils.replace(value,"\\t",""), which means escaping the backslash (\) again.

xliii. If the field is of the string type, replace multiple characters in the character string at a time. For example, replace **h** in **hello** with **j** and **o** with **y** to obtain **jelly**.

Expression: StringUtils.replaceChars(value,"*ho*","*jy*")

xliv. If the string starts with the specified prefix (case sensitive), **true** is returned; otherwise, **false** is returned. For example, **abcdef** starts with **abc**, so that **true** is returned.

Expression: StringUtils.startsWith(value,"*abc*")

xlv. If the field is of the string type, delete all the specified characters at the beginning and end of the field. the field. For example, delete all **x**, **y**, **z**, and **b** from **abcyx** to obtain **abc**.

Expression: StringUtils.strip(value,"*xyz*b")

xlvi. If the field is of the string type, delete all the specified characters at the end of the field, for example, delete the "abc" string at the end of the field.

Expression: StringUtils.stripEnd(value,*"abc"*)

xlvii. If the field is of the string type, delete all the specified characters at the beginning of the field, for example, delete all spaces at the beginning of the field.

Expression: StringUtils.stripStart(value,*null*)

xlviii. If the field is of the string type, obtain the substring after the specified position (the index starts from 0, including the character at the specified position) of the character string. If the specified position is a negative number, calculate the position in the descending order. The first digit at the end is -1. For example, obtain the character whose index is 2 from **abcde** (that is, **c**) and the string after it, that is, **cde**.

Expression: StringUtils.substring(value,*2*)

xlix. If the field is of the string type, obtain the substring in a specified range (the index starts from 0, including the character at the start and excluding the character at the end). If the range is a negative number, calculate the position in the descending order. The first digit at the end is -1. For example, obtain the string between the second character (c) and fourth character (e) of **abcde**, that is, **cd**.

Expression: StringUtils.substring(value,*2*,4)

l. If the field is of the string type, obtain the substring after the first specified character. For example, obtain the substring after the first **b** in **abcba**, that is, **cba**.

Expression: StringUtils.substringAfter(value,"*b*")

li. If the field is of the string type, obtain the substring after the last specified character. For example, obtain the substring after the last **b** in **abcba**, that is, **a**.

Expression: StringUtils.substringAfterLast(value,"*b*")

lii. If the field is of the string type, obtain the substring before the first specified character. For example, obtain the substring before the first **b** in **abcba**, that is, **a**.

Expression: StringUtils.substringBefore(value,"*b*")

liii. If the field is of the string type, obtain the substring before the last specified character. For example, obtain the substring before the last **b** in **abcba**, that is, **abc**.

Expression: StringUtils.substringBeforeLast(value,"*b*")

liv. If the field is of the string type, obtain the substring nested within the specified string. If no substring is found, **null** is returned. For example, obtain the substring between **tag** in **tagabctag**, that is, **abc**.

Expression: StringUtils.substringBetween(value,"*tag*")

lv. If the field is of the string type, delete the control characters (char≤32) at both ends of the character string, for example, delete the spaces at both ends of the character string.

Expression: StringUtils.trim(value)

lvi. Convert the character string to a value of the byte type. If the conversion fails, **0** is returned.

Expression: NumberUtils.toByte(value)

lvii. Convert the character string to a value of the byte type. If the conversion fails, the specified value, for example, **1**, is returned.

Expression: NumberUtils.toByte(value,*1*)

lviii. Convert the character string to a value of the double type. If the conversion fails, **0.0d** is returned.

Expression: NumberUtils.toDouble(value)

lix. Convert the character string to a value of the double type. If the conversion fails, the specified value, for example, **1.1d**, is returned.

Expression: NumberUtils.toDouble(value,*1.1d*)

lx. Convert the character string to a value of the float type. If the conversion fails, **0.0f** is returned.

Expression: NumberUtils.toFloat(value)

lxi. Convert the character string to a value of the float type. If the conversion fails, the specified value, for example, **1.1f**, is returned.

Expression: NumberUtils.toFloat(value,*1.1f*)

lxii. Convert the character string to a value of the int type. If the conversion fails, **0** is returned.

Expression: NumberUtils.toInt(value)

lxiii. Convert the character string to a value of the int type. If the conversion fails, the specified value, for example, **1**, is returned.

Expression: NumberUtils.toInt(value,*1*)

lxiv. Convert the character string to a value of the long type. If the conversion fails, **0** is returned.

Expression: NumberUtils.toLong(value)

lxv. Convert the character string to a value of the long type. If the conversion fails, the specified value, for example, **1L**, is returned.

Expression: NumberUtils.toLong(value,*1L*)

lxvi. Convert the character string to a value of the short type. If the conversion fails, **0** is returned.

Expression: NumberUtils.toShort(value)

lxvii. Convert the character string to a value of the short type. If the conversion fails, the specified value, for example, **1**, is returned.

Expression: NumberUtils.toShort(value,*1*)

lxviii. Convert the IP string to a value of the long type, for example, convert **10.78.124.0** to **172915712**.

Expression: CommonUtils.ipToLong(value)

lxix. Read an IP address and physical address mapping file from the network, and download the mapping file to the map collection. *url* indicates the address for storing the IP mapping file, for example, **http://10.114.205.45:21203/sqoop/IpList.csv**.

Expression: HttpsUtils.downloadMap("*url*")

lxx. Cache the IP address and physical address mappings and specify a key for retrieval, for example, **ipList**.

Expression: CommonUtils.setCache("*ipList*",HttpsUtils.downloadMap("*url*"))

lxxi. Obtain the cached IP address and physical address mappings.

Expression: CommonUtils.getCache("*ipList*")

lxxii. Check whether the IP address and physical address mappings are cached.

Expression: CommonUtils.cacheExists("*ipList*")

lxxiii. Based on the specified offset type (month/day/hour/minute/second) and offset (positive number indicates increase and negative number indicates decrease), convert the time in the specified format to a new time, for example, add 8 hours to **2019-05-21 12:00:00**.

Expression: DateUtils.getCurrentTimeByZone("*yyyy-MM-dd HH:mm:ss*",value, "*hour*", *8*)

lxxiv. If the value is empty or null, "aaa" is returned. Otherwise, **value** is returned.

Expression: StringUtils.defaultIfEmpty(value,*"aaa"*)

Special Links

- If the source link is a DLI link, and the destination link is a DWS link, fields of the tinyint type of the DLI link are mapped to fields of the smallint type of the DWS link.
- If the source link is a Hudi link, and the destination link is a DWS link, fields of the Double type of the Hudi link are mapped to fields of the Float type of the DWS link.

4.6.6 Scheduling Job Execution

CDM supports scheduled execution of table/file migration jobs by minute, hour, day, week, and month. This section describes how to configure scheduled job parameters.

NOTE

- When configuring scheduled jobs, do not set the same scheduled time for different jobs. Instead, set different times to avoid exceptions.
- If you use DataArts Studio DataArts Factory to schedule the CDM migration job and configure this parameter, both configurations take effect. To ensure unified service logic and avoid scheduling conflicts, enable job scheduling in DataArts Factory and do not configure a scheduled task for the job in DataArts Migration.
- The scheduled execution function uses the Java Quartz timer, which is similar to the Cron expression configuration. It parses the minute, hour, day, and month of the start time, and constructs a cronb expression.

  For example, in the daily scheduling mode where the interval is set to 1 day: if the current time is 2022-10-14 12:00 and the start time is set to 2022-10-14 00:00, the job is executed at 2022-10-15 00:00; if the current time is 2022-10-14 12:00 and the start time is set to 2022-10-14 00:00, the job is executed at 2022-10-15 00:00.

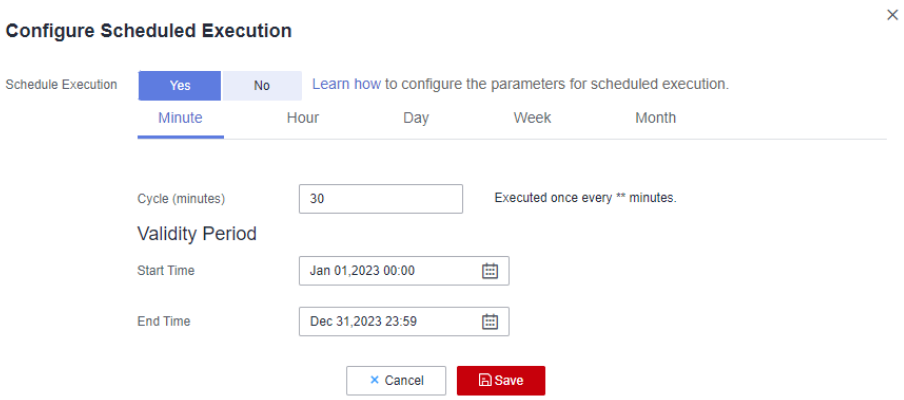
  In the daily scheduling mode where the interval is set to 2 days: if the current time is 2022-10-14 12:00 and the start time is set to 2022-10-14 00:00, the job is executed at 2022-10-16 00:00; if the current time is 2022-10-14 12:00 and the start time is set to 2022-10-14 00:00, the job is executed at 2022-10-16 00:00.

Scheduling Job Execution by Minute

CDM allows jobs to be executed every several minutes. It is recommended that the cycle be at least 5 minutes.

- **Start Time**: indicates the time when the scheduled configuration takes effect, or the first time when the job is automatically executed.
- **Cycle (minutes)**: indicates the interval when a job is executed starting from the start time.
- **End Time**: This parameter is optional. If it is not set, the scheduled job keeps being automatically executed. If it is set, the scheduled job will be automatically stopped at the end time.

Figure 4-43 Scheduling job execution by minute



For example, the settings shown in the above figure mean that the job will be automatically executed at 00:00 on January 1, 2023 for the first time at a cycle of 30 minutes until 23:59 on December 31, 2023.

Scheduling Job Execution by Hour

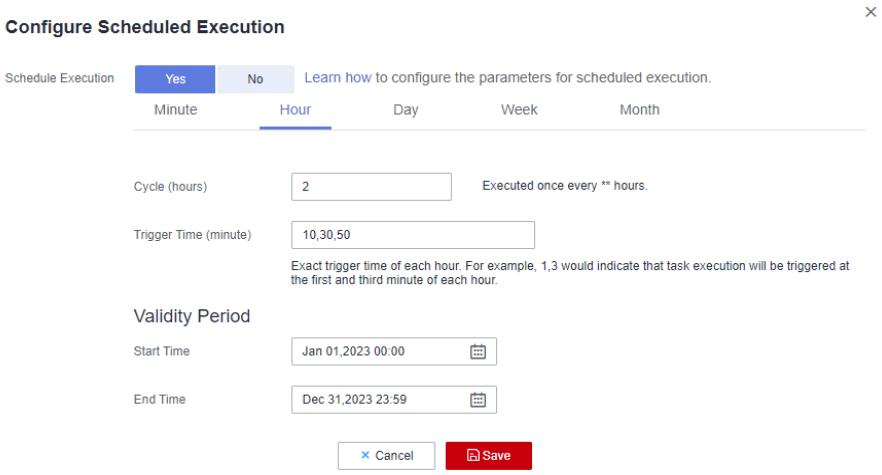
CDM allows jobs to be executed every several hours.

- **Cycle (hours)**: indicates the interval when a job is automatically executed.
- **Trigger Time (minute)**: indicates the exact time in each hour when a scheduled task is triggered. The value ranges from 0 to 59. You can set a maximum of 60 values and use commas (,) to separate these values. However, the values must be unique.

  If the trigger time is not within the validity period, the system selects a trigger time closest to the validity period for the scheduled job to be automatically executed at the first time. The following gives an example:

  - **Start Time**: **1:20**
  - **Cycle (hours)**: **3**
  - **Trigger Time (minute)**: **10**
- **Validity Period**: includes **Start Time** and **End Time**.
  - **Start Time**: indicates the time when the scheduled configuration takes effect.
  - **End Time**: This parameter is optional, which indicates the time when the scheduled job is automatically stopped. If this parameter is not set, the scheduled job keeps being automatically executed.

Figure 4-44 Scheduling job execution by hour



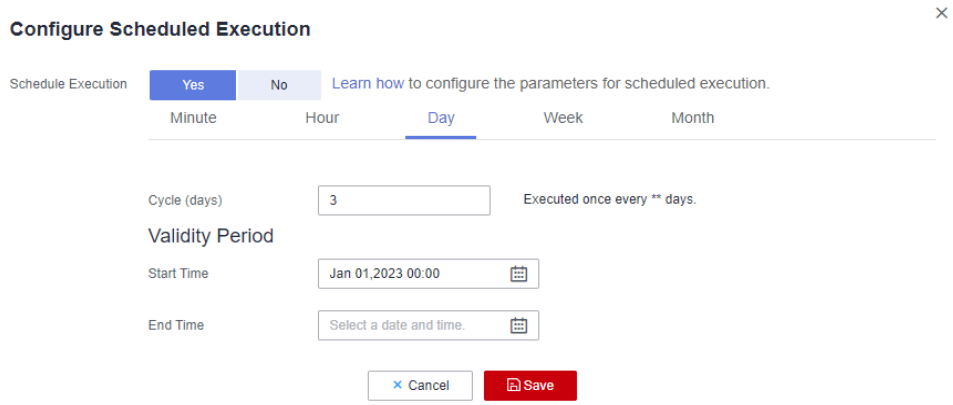
For example, the settings shown in the above figure mean that the job will be automatically executed at 00:10 on January 1, 2023 for the first time, at 00:30 for the second time, and at 00:50 for the third time. It will be executed three times every two hours until 23:59 on December 31, 2023.

Scheduling Job Execution by Day

CDM allows jobs to be executed every several days.

- **Cycle (days)**: indicates the interval when a job is executed starting from the start time.
- **Validity Period**: includes **Start Time** and **End Time**.
  - **Start Time**: indicates the time when the scheduled configuration takes effect, or the first time when the job is automatically executed.
  - **End Time**: This parameter is optional, which indicates the time when the scheduled job is automatically stopped. If this parameter is not set, the scheduled job keeps being automatically executed.

Figure 4-45 Scheduling job execution by day



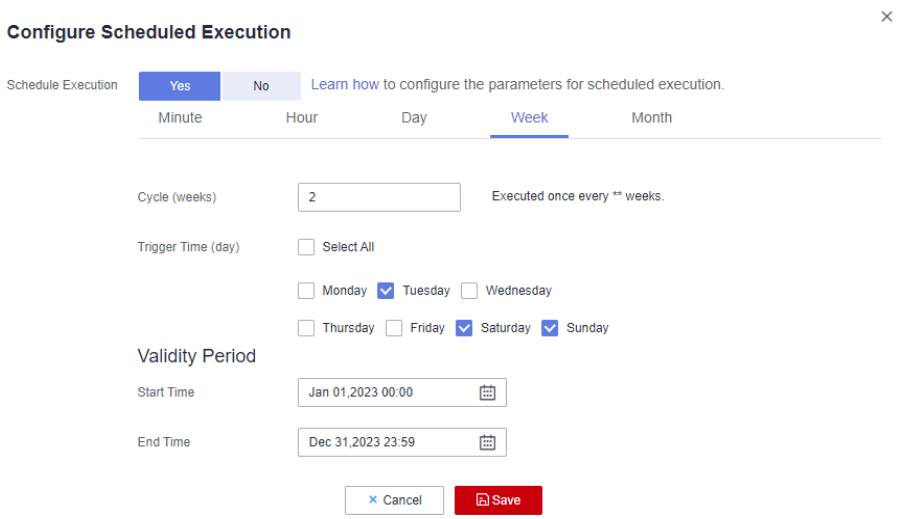
For example, the settings shown in the above figure mean that the job will be automatically executed at 00:00 on January 1, 2023 for the first time, and will be executed once every three days. The configuration is valid permanently.

Scheduling Job Execution by Week

CDM allows jobs to be executed every several weeks.

- **Cycle (weeks)**: indicates the interval when a scheduled job is executed starting from the start time.
- **Trigger Time (day)**: You can specify the day of each week when the job is automatically executed. One or more days can be selected at a time.
- **Validity Period**: includes **Start Time** and **End Time**.
  - **Start Time**: indicates the time when the scheduled configuration takes effect.
  - **End Time**: This parameter is optional, which indicates the time when the scheduled job is automatically stopped. If this parameter is not set, the scheduled job keeps being automatically executed.

Figure 4-46 Scheduling job execution by week



For example, the settings shown in the above figure mean that the job will be automatically executed at 00:00 every Tuesday, Saturday, and Sunday every two weeks starting from 00:00 on January 1, 2023 until 23:59 on December 31, 2023.

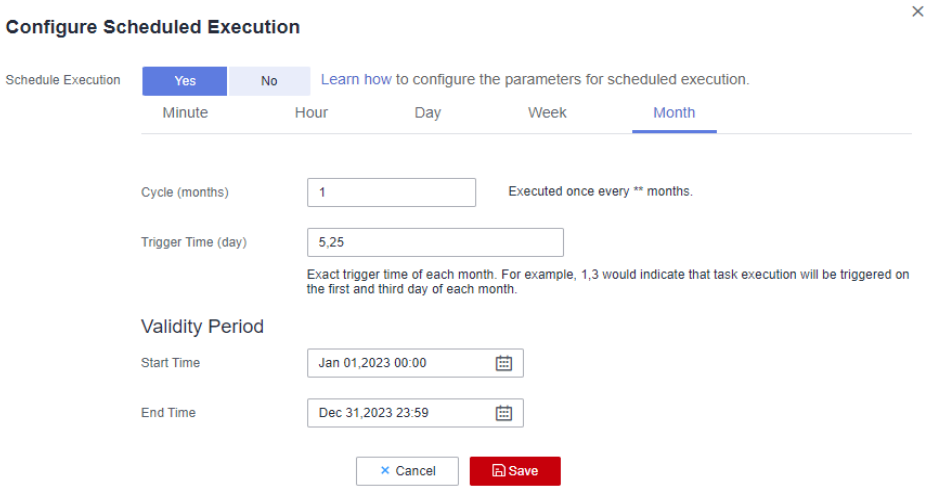
Scheduling Job Execution by Month

CDM allows jobs to be executed every several months.

- **Cycle (months)**: indicates the interval when a scheduled job is executed starting from the start time.
- **Trigger Time (day)**: indicates the day of each month when the job is executed. The value ranges from 1 to 31. You can set multiple values and use commas (,) to separate these values. However, the values must be unique.

- **Validity Period**: includes **Start Time** and **End Time**.
  - **Start Time**: indicates the time when the scheduled configuration takes effect. The automatic execution time is accurate to hour, minute, and second.
  - **End Time**: This parameter is optional, which indicates the time when the scheduled job is automatically stopped. If this parameter is not set, the scheduled job keeps being automatically executed.

Figure 4-47 Scheduling job execution by month



For example, the settings shown in the above figure mean that the job will be automatically executed at 00:00 on the 5th and 25th days of each month starting from 00:00 on January 1, 2023 until 23:59 on December 31, 2023.

4.6.7 Job Configuration Management

On the **Settings** tab page, you can perform the following operations:

- **Maximum Concurrent Extractors**
- **Scheduled Backup/Restoration**
- **Environment Variables of Job Parameters**

Maximum Concurrent Extractors

Maximum number of concurrent extraction tasks in a cluster

NOTE

This parameter is also available on the **Cluster Configuration** page. You can change its value either on this page or the **Cluster Configuration** page.

CDM migrates data through data migration jobs. It works in the following way:

1. When data migration jobs are submitted, CDM splits each job into multiple tasks based on the **Concurrent Extractors** parameter in the job configuration.

NOTE

Jobs for different data sources may be split based on different dimensions. Some jobs may not be split based on the **Concurrent Extractors** parameter.

2. CDM submits the tasks to the running pool in sequence. Tasks (defined by **Maximum Concurrent Extractors**) run concurrently. Excess tasks are queued.

By setting appropriate values for the **Concurrent Extractors** and **Maximum Concurrent Extractors** parameters, you can accelerate migration.

1. You are advised to set **Maximum Concurrent Extractors** to twice the number of vCPUs. For details, see **Table 4-98**.

   Table 4-98 Recommended maximum number of concurrent extractors for a CDM cluster

Flavor	vCPUs/Memory	Recommended Maximum Concurrent Extractors
cdm.large	8 vCPUs, 16 GB	16
cdm.xlarge	16 vCPUs, 32 GB	32
cdm.4xlarge	64 vCPUs, 128 GB	128

2. Configure the number of concurrent extractors based on the following rules:
   a. When data is to be migrated to files, CDM does not support multiple concurrent tasks. In this case, set a single process to extract data.
   b. If each row of the table contains less than or equal to 1 MB data, data can be extracted concurrently. If each row contains more than 1 MB data, it is recommended that data be extracted in a single thread.
   c. Set **Concurrent Extractors** for a job based on **Maximum Concurrent Extractors** for the cluster. It is recommended that **Concurrent Extractors** is less than **Maximum Concurrent Extractors**.
   d. If the destination is DLI, you are advised to set the number of concurrent extractors to 1. Otherwise, data may fail to be written.

Scheduled Backup/Restoration

This function depends on the OBS service.

- Prerequisites

  An OBS link has been created. For details, see **Link to OBS**.
- Scheduled backup

  On the **Job Management** page, click **Settings** and configure **Scheduled Backup** and its related parameters.

Table 4-99 Scheduled backup parameters

Parameter	Description	Example Value
Scheduled Backup	Whether to enable automatic backup. This function is used to back up jobs but not links.	Enable
Backup Policy	• **All jobs**: CDM backs up all table/file migration jobs and entire DB migration jobs regardless of the job statuses. However, historical jobs are not backed up. • **All jobs by groups**: You select one or more job groups to back up.	All jobs
Backup Cycle	Select the backup cycle. • **Day**: The backup is performed daily at 00:00:00. • **Week**: The backup is performed at 00:00:00 every Monday. • **Month**: The backup is performed at 00:00:00 on the first day of each month.	Day
OBS Link for Writing Backups	Link used to back up jobs to OBS buckets. Select a link you have created on the **Links** page.	obslink
OBS Bucket	OBS bucket where backup files are stored	cdm
Backup Data Directory	Directory where backup files are stored	/cdm-bk/

- Restoring jobs

  If automatic backup has been performed, the backup list is displayed on the **Configuration Management** tab page. The OBS buckets where the backup files reside, backup paths, and backup time are displayed.

  You can click **Restore Backup** in the **Operation** column of the backup list to restore the CDM jobs.

Environment Variables of Job Parameters

When creating a migration job on CDM, the parameter (such as the OBS bucket name or file path) that can be manually configured, a field in a parameter, or a character in a field can be configured as a global variable, so that you can change parameter values in batches, or batch replace certain characters after jobs are exported or imported.

The following describes how to batch replace the OBS bucket name in a migration job.

1. On the **Job Management** page, click the **Configuration Management** tab and configure environment variables.

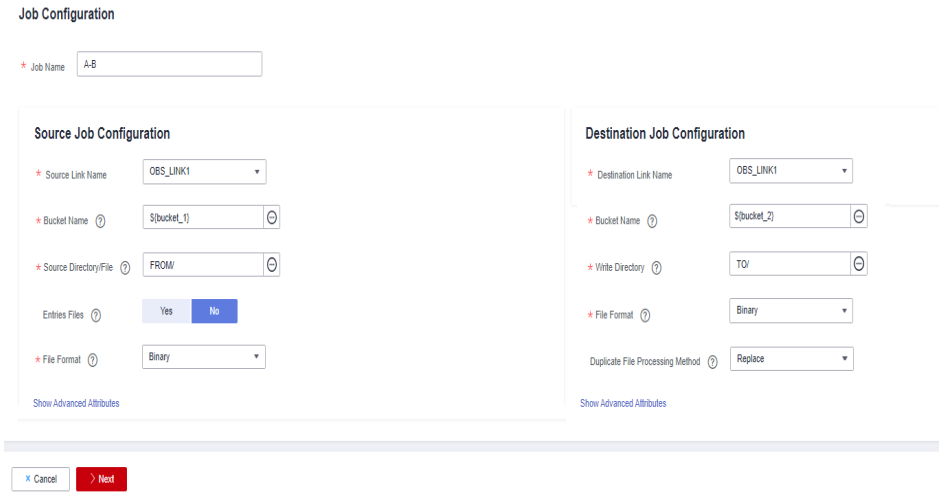

```
bucket_1=A
bucket_2=B
```

Variable **bucket_1** indicates bucket A, and variable **bucket_2** indicates bucket B.

2. On the page for creating a CDM migration job, migrate data from bucket A to bucket B.

   Set the source bucket name to **${bucket_1}** and destination bucket name to **${bucket_2}**.

   Figure 4-48 Setting the bucket names to environment variables



3. If you want to migrate data from bucket C to bucket D, you do not need to change the job parameters. You only need to change the environment variables on the **Configuration Management** tab page as follows:

```
bucket_1=C
bucket_2=D
```

4.6.8 Managing a Single Job

Existing CDM jobs can be viewed, modified, deleted, started, and stopped. This section describes how to view and modify a job.

Viewing a Job

- **Viewing job status**

  The job status can be **New**, **Pending**, **Booting**, **Running**, **Failed**, **Succeeded**, or **Stopped**.

  Pending indicates that the job is waiting to be scheduled by the system, and **Booting** indicates that the data to be migrated is being analyzed.

- **Viewing the historical records**

  On the **Historical Record** page, you can view job execution records, read/write statistics, and job execution logs.

- **Viewing job logs**

  On the **Historical Record** page, you can view all logs of a job.

  Alternatively, in the **Operation** column, choose **More** > **Log** to view the latest logs of the job.

- **Viewing the JSON file of a job**

  You can directly edit the JSON file of a job, which is equivalent to modifying the parameter settings of the job.

- **Querying the job statistics**

  You can open the preview window of a configured database job and view up to 1,000 pieces of data. By comparing the number of data records of the migration source and destination, you can check whether the migration was successful and whether data was lost.

Modifying a Job

- **Modifying the job parameters**

  You can reconfigure job parameters, but you cannot reselect source and destination links.

- **Editing the JSON file of a job**

  You can directly edit the JSON file of a job, which is equivalent to modifying the parameter settings of the job.

Procedure

Step 1 Log in to the management console and choose **Service List** > **Cloud Data Migration**. In the left navigation pane, choose **Cluster Management**. Locate the target cluster and click **Job Management**.

Step 2 Click **Historical Jobs** to view all historical jobs executed in the latest month.

Step 3 Click **Table/File Migration**. The job list is displayed. You can perform the following operations on a single job:

- Modify the job parameters: Click **Edit** in the **Operation** column to modify the job parameters.
- Run the job: Click **Run** in the **Operation** column to manually start the job.
- View the historical records: Click **Historical Record** in the **Operation** column. On the **Historical Record** page that is displayed, view the job's historical execution records and read/write statistics. Click **Log** to view the job logs.
- Delete the job: Choose **More** > **Delete** in the **Operation** column to delete the job.
- Stop the job: Choose **More** > **Stop** in the **Operation** column to stop the job.
- View the job JSON: Choose **More** > **View Job JSON** in the **Operation** column to view the job JSON.
- Edit the job JSON: Choose **More** > **Edit Job JSON** in the **Operation** column to edit the job JSON files, which is similar to modify the job parameters.
- Configure a scheduled job: Locate a job and choose **More** > **Configure Scheduled Execution**. You can set the cycle for periodically executing the job. For details, see **Scheduling Job Execution**.
- View logs: Locate a job, click **More** in the **Operation** column, and select **Log** to view the latest log of the job.

  You can also view all logs of the job on the **Historical Record** page.
- Retry the job: Locate a failed job, click **More** in the **Operation** column, and select **Retry**. The job will be automatically retried three times.

Step 4 After the modification, click **Save** or **Save and Run**.

----End

4.6.9 Managing Jobs in Batches

Scenario

This section describes how to manage CDM table/file migration jobs in batches. The following operations are involved:

- Manage jobs by group.
- Run jobs in batches.
- Delete jobs in batches.
- Export jobs in batches.
- Import jobs in batches.

You can export and import jobs in batches in the following scenarios:

- Job migration between CDM clusters: You can migrate jobs from a cluster of an earlier version to a new version.
- Job backup: You can stop or delete CDM clusters to reduce costs. In this case, you can export the job scripts in batches and save them, and create a cluster and import the job scripts if necessary.
- Batch job creation: You can manually create a job and export the job configuration file in JSON format. Copy the content in the JSON file to the same file or new files, and then import the file/files to CDM to create jobs in batches.

Procedure

Step 1 Log in to the management console and choose **Service List** > **Cloud Data Migration**. In the left navigation pane, choose **Cluster Management**. Locate the target cluster and click **Job Management**.

Step 2 Click **Table/File Migration**. The job list is displayed. You can perform the following batch operations:

- **Manage jobs by group.**

  CDM allows users to add, modify, search for, and delete job groups. When a group is deleted, all jobs in the group are deleted.

  When creating a job, if jobs have been assigned to different groups, you can display, start, or export jobs by group.

  NOTE

  Starting jobs by group will run all jobs in the group. If user isolation is enabled, starting jobs by group will still run all jobs in the group even if otherIAM users in the a Huawei account cannot view the jobs in the group. Therefore, you are not advised to start jobs by group in user isolation scenarios.

- **Run jobs in batches.**

  After selecting one or more jobs, click **Run** to start these jobs in batches.

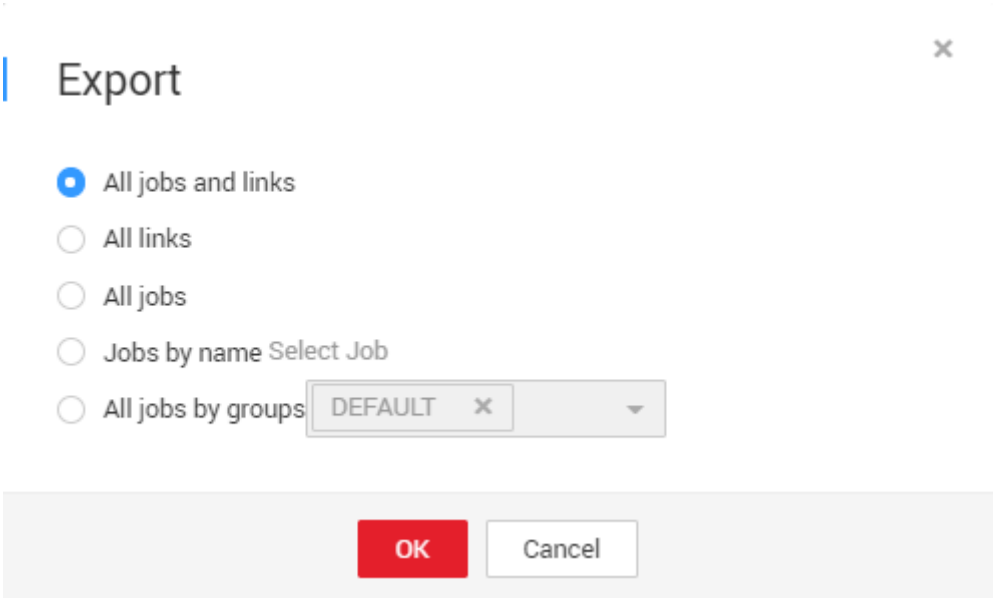
- **Delete jobs in batches.**

After selecting one or more jobs, click **Delete** to delete these jobs in batches.

- **Export jobs in batches.**

  Click **Export**.

  Figure 4-49 Export

  - **All jobs and links**: Export all jobs and links at a time.
  - **All jobs**: Export all jobs at a time.
  - **All links**: Export all links at a time.
  - **Jobs by name**: Select the jobs to export and click **OK**.
  - **All jobs by groups**: Select the group to export and click **OK**.

  Exported jobs are stored in JSON files, which can be used as backups or imported to other clusters.

  NOTE

  For security purposes, no link password is exported when jobs are exported. All passwords are replaced by *Add password here*.

- **Import jobs in batches.**

  Click **Import** and select the import format (text file or JSON).

  - **By JSON string**: Job files to be imported must be in JSON format and the file size cannot exceed 1 MB. If the job files to be imported are exported from CDM, edit the JSON files before importing them to CDM. Replace *Add password here* with the correct link passwords.
  - **By text file**: This mode can be used when the local JSON files cannot be uploaded properly. Paste the JSON strings for the jobs into the text box.

  NOTE

  Existing jobs cannot be overwritten during the import.

----End

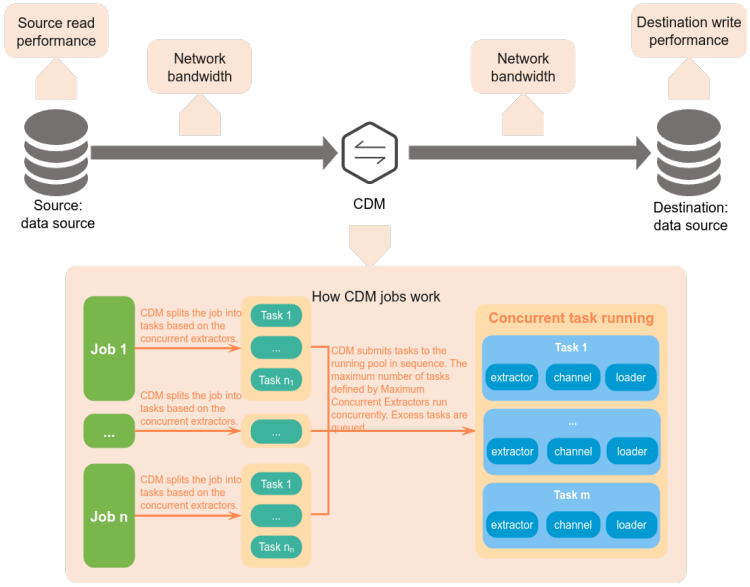
4.7 Improving Migration Performance

4.7.1 How Migration Jobs Work

Data Migration Model

Figure 4-50 shows the simplified migration model used by CDM.

Figure 4-50 Migration model used by CDM



CDM migrates data through data migration jobs. It works in the following way:

1. When data migration jobs are submitted, CDM splits each job into multiple tasks based on the **Concurrent Extractors** parameter in the job configuration.

   NOTE

   Jobs for different data sources may be split based on different dimensions. Some jobs may not be split based on the **Concurrent Extractors** parameter.

2. CDM submits the tasks to the running pool in sequence. Tasks (defined by **Maximum Concurrent Extractors**) run concurrently. Excess tasks are queued.

Factors Affecting Migration Performance

According to the migration model, the migration speed is affected by factors such as the source read speed, network bandwidth, destination write performance, and CDM cluster and job configuration.

Table 4-100 Factors affecting migration performance

Factor		Description
Service-related factors	Concurrent extractors of a job	The number of concurrent extractors can be set for a CDM job during the job creation. Setting a proper value for this parameter can effectively improve the migration speed. If the value is too small, migration will be too slow. If the value is too large, the migration job is overloaded and may fail. • When data is to be migrated to files, CDM does not support multiple concurrent tasks. In this case, set a single process to extract data. • If each row of the table contains less than or equal to 1 MB data, data can be extracted concurrently. If each row contains more than 1 MB data, it is recommended that data be extracted in a single thread.
	Maximum concurrent extractors of a cluster	Setting a proper value for this parameter can effectively improve the migration speed. If the value is too small, migration will be too slow. If the value is too large, the source is overloaded and the system may be unstable. The maximum concurrent extractors vary depending on the CDM cluster flavor. The upper limit is twice the number of vCPUs. The following are the maximum concurrent extractors of some flavors: • cdm.large: 16 • cdm.xlarge: 32 • cdm.4xlarge: 128
	Service model	If the number of CDM jobs that run concurrently exceeds the maximum concurrent extractors for the CDM cluster, some jobs will be queued, and the migration will be prolonged. Avoid running too many jobs simultaneously, which may cause slow migration due to insufficient resources.
	Data model	The migration speed is also affected by the data structure. The following are some examples: • The wider a table is and the more string types the table has, the slower the migration is. • A large file is migrated more quickly than multiple small files whose total size is the same as the large file. • The more content a message has and the higher bandwidth it uses, the less transactions per second (TPS) are.
Source read speed		It depends on the performance of the data source at the source. For details about how to increase the read speed, see the documents of data sources at the source.

Factor	Description
Network bandwidth	The CDM cluster can communicate with the data source through an intranet, public network VPN, NAT, or Direct Connect. • If they communicate through an intranet, the network bandwidth varies depending on the CDM instance flavor. – For cdm.large instances, the baseline and maximum bandwidths of the CDM cluster NIC are 0.8 and 3 Gbit/s, respectively. – For cdm.xlarge instances, the baseline and maximum bandwidths of the CDM cluster NIC are 4 and 10 Gbit/s, respectively. – For cdm.4xlarge instances, the baseline and maximum bandwidths of the CDM cluster NIC are 36 and 40 Gbit/s, respectively. • If they communicate through the Internet, the network bandwidth is subject to the Internet bandwidth. The bandwidth for the CDM cluster depends on the EIP bound to the CDM cluster, and the bandwidth for the data source depends on the Internet bandwidth. • If they communicate through a VPN, NAT, or Direct Connect, the network bandwidth is subject to the VPN, NAT, or Direct Connect bandwidth.
Destination write performance	It depends on the performance of the data source at the destination. For details about how to improve the performance, see the documents of data sources at the destination.

4.7.2 Performance Tuning

Overview

In addition to increasing the source read speed, improving the destination write performance, and increasing the bandwidth, you can accelerate migration using the following methods:

- **Use a CDM cluster of higher specifications**

  The NIC bandwidth and maximum number of concurrent extractors vary depending on the CDM cluster specifications. If you want to migrate data faster, or the metrics of your CDM cluster (such as the CPU usage, disk usage, and memory usage) are often high, you may need a CDM cluster with higher specifications for data migration.

- **Use multiple CDM clusters**

  In some scenarios, you are advised to use multiple CDM clusters to share workloads to improve migration efficiency and stability. The following are some examples:

  - Multiple CDM clusters are required for different purposes or by multiple business departments. For example, you may need one CDM cluster for running data migration jobs and another one as an agent for DataArts Studio Management Center.
  - You want to migrate a large number of tables. In this case, you can use multiple CDM clusters to run jobs simultaneously to improve migration efficiency.
  - The CPU usage, disk usage, and memory usage of the in-use CDM cluster are often high. In this case, you are advised to use multiple CDM clusters to shared workloads.

- **Avoid running too many CDM jobs simultaneously**

  If the number of CDM jobs that run concurrently exceeds the maximum concurrent extractors for the CDM cluster, some jobs will be queued, and the migration will be prolonged.

  Avoid running too many jobs simultaneously, which may cause slow migration due to insufficient resources.

- **Change concurrent extractors**

  If the number of tasks is small, adjusting the number of concurrent extractors is the best way to improve performance. You can set the number of concurrent extractors for a job and the maximum number of concurrent extractors for a cluster.

  CDM migrates data through data migration jobs. It works in the following way:

  a. When data migration jobs are submitted, CDM splits each job into multiple tasks based on the **Concurrent Extractors** parameter in the job configuration.

     NOTE

     Jobs for different data sources may be split based on different dimensions. Some jobs may not be split based on the **Concurrent Extractors** parameter.

  b. CDM submits the tasks to the running pool in sequence. Tasks (defined by **Maximum Concurrent Extractors**) run concurrently. Excess tasks are queued.

  By setting appropriate values for parameters **Concurrent Extractors** and **Maximum Concurrent Extractors**, you can accelerate migration. For details about how to change **Concurrent Extractors**, see **Changing Concurrent Extractors**.

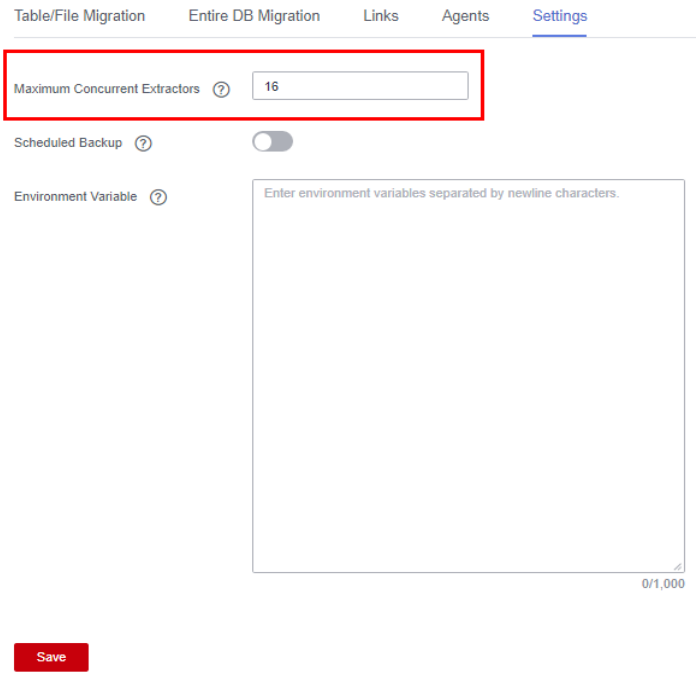
Changing Concurrent Extractors

1. The maximum number of concurrent extractors for a cluster varies depending on the CDM cluster flavor. You are advised to set the maximum number of concurrent extractors to twice the number of vCPUs of the CDM cluster.

Table 4-101 Maximum number of concurrent extractors for a CDM cluster

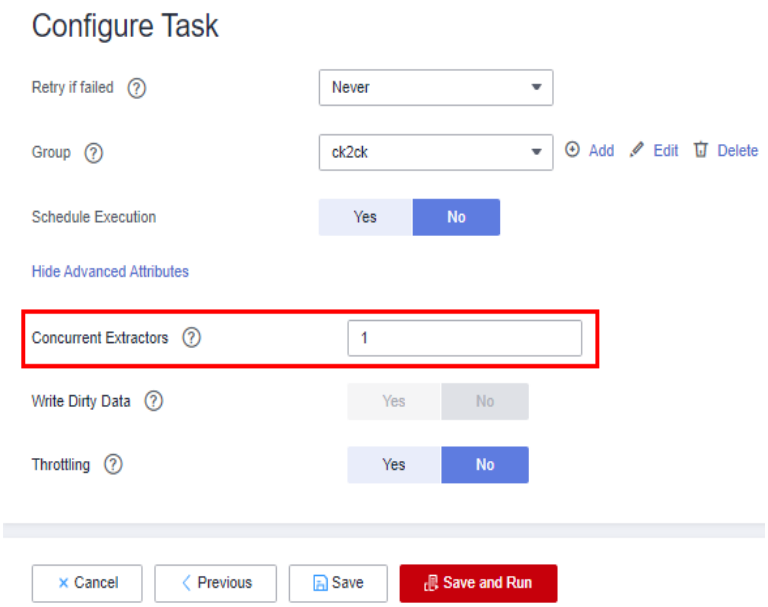
Flavor	vCPUs/Memory	Maximum Concurrent Extractors
cdm.large	8 vCPUs, 16 GB	16
cdm.xlarge	16 vCPUs, 32 GB	32
cdm.4xlarge	64 vCPUs, 128 GB	128

Figure 4-51 Setting Maximum Concurrent Extractors for a CDM cluster



2. Configure the number of concurrent extractors based on the following rules:
   a. When data is to be migrated to files, CDM does not support multiple concurrent tasks. In this case, set a single process to extract data.
   b. If each row of the table contains less than or equal to 1 MB data, data can be extracted concurrently. If each row contains more than 1 MB data, it is recommended that data be extracted in a single thread.
   c. Set **Concurrent Extractors** for a job based on **Maximum Concurrent Extractors** for the cluster. It is recommended that **Concurrent Extractors** is less than **Maximum Concurrent Extractors**.
   d. If the destination is DLI, you are advised to set the number of concurrent extractors to 1. Otherwise, data may fail to be written.

Figure 4-52 Setting Concurrent Extractors for a job



4.7.3 Reference: Job Splitting Dimensions

CDM splits jobs for different data sources based on different dimensions. **Table 4-102** lists the splitting dimensions.

Table 4-102 Job splitting dimensions for different data sources

Data Source Category	Data Source	Job Splitting Rule
Data warehouse	GaussDB(DWS)	• Jobs can be split based on table fields. • Jobs cannot be split based on table partitions.
	Data Lake Insight (DLI)	• Jobs can be split based on the partitioning information of partitioned tables. • Jobs cannot be split based on non-partitioned tables.
Hadoop	MRS HDFS	Jobs can be split based on files.
	MRS HBase	Jobs can be split based on HBase regions.
	MRS Hive	• When the read mode is HDFS, jobs can be split based on Hive files. • When the read mode is JDBC, jobs cannot be split.
	FusionInsight HDFS	Jobs can be split based on files.

Data Source Category	Data Source	Job Splitting Rule
	FusionInsight HBase	Jobs can be split based on HBase regions.
	FusionInsight Hive	• When the read mode is HDFS, jobs can be split based on Hive files. • When the read mode is JDBC, jobs cannot be split.
	Apache HDFS	Jobs can be split based on files.
	Apache HBase	Jobs can be split based on HBase regions.
	Apache Hive	• When the read mode is HDFS, jobs can be split based on Hive files. • When the read mode is JDBC, jobs cannot be split.
Object storage	Object Storage Service (OBS)	Jobs can be split based on files.
File system	FTP	Jobs can be split based on files.
	SFTP	Jobs can be split based on files.
	HTTP	Jobs can be split based on files.
Relational database	RDS for MySQL	• Jobs can be split based on table fields. • Jobs can be split based on table partitions only when **Extract by Partition** is configured.
	RDS for PostgreSQL	• Jobs can be split based on table fields. • Jobs can be split based on table partitions only when **Extract by Partition** is configured.
	RDS for SQL Server	• Jobs can be split based on table fields. • Jobs can be split based on table partitions only when **Extract by Partition** is configured.
	MySQL	• Jobs can be split based on table fields. • Jobs can be split based on table partitions only when **Extract by Partition** is configured.
	PostgreSQL	• Jobs can be split based on table fields. • Jobs can be split based on table partitions only when **Extract by Partition** is configured.

Data Source Category	Data Source	Job Splitting Rule
	Microsoft SQL Server	• Jobs can be split based on table fields. • Jobs cannot be split based on table partitions.
	Oracle	• Jobs can be split based on table fields. • Jobs can be split based on table partitions only when **Extract by Partition** is configured.
	SAP HANA	• Jobs can be split based on table fields. • Jobs cannot be split based on table partitions.
	Database shard	Each backend connects to a subjob, which can be split based on primary keys.
NoSQL	Distributed Cache Service (DCS)	Jobs cannot be split.
	Redis	Jobs cannot be split.
	Document Database Service (DDS)	Jobs cannot be split.
	MongoDB	Jobs cannot be split.
	Cassandra	Jobs can be split based on the token range of Cassandra.
Message system	Data Ingestion Service (DIS)	Jobs can be split based on topics.
	Apache Kafka	Jobs can be split based on topics.
	DMS Kafka	Jobs can be split based on topics.
	MRS Kafka	Jobs can be split based on topics.
Search	Elasticsearch	Jobs cannot be split.
	Cloud Search Service (CSS)	Jobs cannot be split.

4.7.4 Reference: CDM Performance Test Data

Background

The performance metrics provided in this document are for reference only. The performance at your site may be affected by factors such as the data source

performance at the source or destination, network bandwidth, latency, and the data and service model. It is recommended that you test the speed with a small amount of data before migration.

Environment

- An xlarge CDM cluster of the 2.9.1 200 version
- A table which has 50 million rows and 100 columns, and three HDFS binary files which have 35.97 million rows and 100 columns, 66.67 million rows and 100 columns, and 100 million rows and 100 columns, respectively.
- Number of concurrent extraction jobs for determining the maximum extraction/write rate: 1, 10, 20, 30, and 50

Data Source Extraction and Write Performance Test Data

Table 4-103 and **Table 4-104** provide the data extraction and write performance, respectively.

Table 4-103 Data extraction performance

Data Source	Specifications	Version	Extraction Rate for a Single Job (Lines per Second)	Extraction Rate for Multiple Jobs (Lines per Second)
RDS for MySQL	8 vCPUs, 32 GB	MySQL 5.7	42,052	195,313 (concurrency: 40)
Oracle	8 vCPUs, 16 GB	19C	18,539	18,706 (concurrency: 10)
MRS HBase	Master: 16 vCPUs, 64 GB x 3 Node: 8 vCPUs, 32 GB x 3	MRS 3.1.0	6,296	69,156 (concurrency: 30)
MRS Hive	Master: 16 vCPUs, 64 GB x 3 Node: 8 vCPUs, 32 GB x 3	MRS 3.1.0	22,321	170,068 (concurrency: 30)
MRS HDFS (binary files)	Master: 16 vCPUs, 64 GB x 3 Node: 8 vCPUs, 32 GB x 3	MRS 3.1.0	138,727	141,468 (concurrency: 20)
			125,556	126,990 (concurrency: 10)
			120,919	120,919 (concurrency: 10)

Data Source	Specifications	Version	Extraction Rate for a Single Job (Lines per Second)	Extraction Rate for Multiple Jobs (Lines per Second)
DWS	8 vCPUs, 16 GB	8.1.1.300	13,434	/
DLI	16 vCPUs	SQL queue	71,023	19,290 (concurrency: 20)
MRS Hudi (MOR)	Master: 16 vCPUs, 64 GB x 3 Node: 8 vCPUs, 64 GB x 3	MRS 3.2.0	75187	467,289 (concurrency: 30)
MRS Hudi (COW)	Master: 16 vCPUs, 64 GB x 3 Node: 8 vCPUs, 64 GB x 3	MRS 3.2.0	84033	485,436 (concurrency: 30)
ClickHouse	Node: 8 vCPUs, 32 GB x 2	ClickHouse 22.3.2.2	187265	/
Elasticsearch	4 vCPUs, 8 GB x 6	Elasticsearch 7.10.2	28752	/
RDS for PostgreSQL	4 vCPUs, 32 GB (active/standby)	PostgreSQL 13.12	128865	1,351,351 (concurrency: 30)

Table 4-104 Data write performance

Data Source	Specifications	Version	Write Rate for a Single Job (Rows per Second)	Write Rate for Multiple Jobs (Rows per Second)
RDS for MySQL	8 vCPUs, 32 GB	MySQL 5.7	2,658	/
Oracle	8 vCPUs, 16 GB	19C	/	/

Data Source	Specifications	Version	Write Rate for a Single Job (Rows per Second)	Write Rate for Multiple Jobs (Rows per Second)
MRS HBase	Master: 16 vCPUs, 64 GB x 3 Node: 8 vCPUs, 32 GB x 3	MRS 3.1.0	3,959	4,120 (concurrency: 10)
MRS Hive	Master: 16 vCPUs, 64 GB x 3 Node: 8 vCPUs, 32 GB x 3	MRS 3.1.0	25,813	26,882 (concurrency: 10)
MRS HDFS (binary files)	Master: 16 vCPUs, 64 GB x 3 Node: 8 vCPUs, 32 GB x 3	MRS 3.1.0	65,075	90,155 (concurrency: 10)
			86,248	86,248 (concurrency: 1)
			76,687	76,687 (concurrency: 1)
DWS	8 vCPUs, 16 GB	8.1.1.300	26,624	27,902 (concurrency: 10)
DLI	16 vCPUs	SQL queue	15,211	18,430 (concurrency: 10)
MRS Hudi (MOR)	Master: 16 vCPUs, 64 GB x 3 Node: 8 vCPUs, 64 GB x 3	MRS 3.2.0	16345	183,150 (concurrency: 10)
MRS Hudi (COW)	Master: 16 vCPUs, 64 GB x 3 Node: 8 vCPUs, 64 GB x 3	MRS 3.2.0	21088	88,183 (concurrency: 20)
ClickHouse	Node: 8 vCPUs, 32 GB x 2	ClickHouse 22.3.2.2	93984	/
Elasticsearch	4 vCPUs, 8 GB x 6	Elasticsearch 7.10.2	22271	/

Data Source	Specifications	Version	Write Rate for a Single Job (Rows per Second)	Write Rate for Multiple Jobs (Rows per Second)
RDS for PostgreSQL	4 vCPUs, 32 GB (active/standby)	PostgreSQL 13.12	34746	153,374 (concurrency: 10)

4.8 Error Codes

If an exception occurs during the execution of an operation request and the request is not processed, an error message is returned. The error information contains the error code and error description. **Table 4-105** lists some common error code in CDM error messages. You can handle the exceptions by referring to the solutions in **Table 4-105**.

Error Code Description

Table 4-105 Description

Error Code	Error Message	Solution
Cdm.0000	System error.	Contact customer service or technical support.
Cdm.0003	Kerberos login failed.	Check whether the keytab and principal configuration files are correct.
Cdm.0009	*%s* is not an integer or is beyond the value range [0, 2147483647].	Modify the parameter settings based on the error message and try again.
Cdm.0010	The integer must be within the range of [*%s*].	Check whether the parameter value is valid based on the error message. If it is not, correct it and try again.
Cdm.0011	The parameter value exceeds the value range.	Check whether the parameter value is valid based on the error message. If it is not, correct it and try again.
Cdm.0012	JDBC driver class is not found.	Contact customer service or technical support.

Error Code	Error Message	Solution
Cdm.0013	Failed to connect to the agent.	It is possible that the network is disconnected, or no security group or firewall rule is configured to allow access. If the fault persists, contact customer service or technical support.
Cdm.0014	The parameter is invalid.	Change the parameter value and try again.
Cdm.0015	An error occurred during file parse.	Check whether the content or format of the uploaded file is correct. If it is not, correct it and try again.
Cdm.0016	The file to be uploaded cannot be empty.	Ensure that the file you uploaded is not empty and try again.
Cdm.0017	MRS Kerberos authentication failed.	Check whether the password used for Kerberos authentication is strong. If it is not, change to a strong password and try again.
Cdm.0018	The content of jobs or links is invalid.	Contact customer service or technical support.
Cdm.0019	Invalid IP address and port number.	Try again later or contact customer service or technical support.
Cdm.0020	The string must contain the following substring: *%s*.	Modify the parameter settings based on the error message and try again.
Cdm.0021	Failed to connect to the server: *%s*.	Contact customer service or technical support.
Cdm.0023	Failed to write data. Cause: *%s*.	Contact customer service or technical support.
Cdm.0024	[*%s*] must be within the range of [*%s*].	Modify the parameter settings based on the error message and try again.
Cdm.0025	The length of the written data exceeds the length defined by the table field. Error message: %s.	Modify the length of the data to be written based on the error message and try again.
Cdm.0026	The primary key already exists. Error message: %s.	Check the data based on the error message and resolve the primary key conflict.

Error Code	Error Message	Solution
Cdm.0027	The code of the written character string may be different from the code defined in the table. Error message: %s.	Modify the character string code based on the error message.
Cdm.0028	Incorrect username or password. Error message: %s.	Change the username or password and try again.
Cdm.0029	The database name does not exist. Error message: %s.	Select a correct database and try again.
Cdm.0030	Incorrect username, password, or database name. Error message: %s.	Correct the username, password, and database name as prompted and try again.
Cdm.0031	The connection timed out.	Connection timed out. Check whether the IP address, host name, and port number are correct, and whether the security group and firewall are correctly configured.
Cdm.0032	Incorrect username or password. See the error message returned by the server: %s.	Change the username and password based on the error message and try again.
Cdm.0033	SIMPLE authentication is not supported.	Select the Kerberos authentication type and try again.
Cdm.0034	Restart the CDM cluster to reload MRS or FusionInsight configurations.	Restart the CDM cluster to reload MRS or FusionInsight configurations.
Cdm.0035	You do not have the write permission on the file. Error message: %s.	Configure the permission based on the error message and try again.
Cdm.0036	Invalid datestamp or date format. Error message: %s.	Configure the datestamp or date format based on the error message and try again.
Cdm.0037	The parameter is invalid. Error message: %s.	Modify the parameter settings based on the error message and try again.
Cdm.0038	The connection timed out.	Check the VPC and security group rules.
Cdm.0039	The connection name cannot be modified.	The connection name cannot be changed.

Error Code	Error Message	Solution
Cdm.0040	Logs are deleted because they are periodically cleared.	Contact customer service or technical support.
Cdm.0041	The group in use cannot be updated or deleted.	Do not modify the group.
Cdm.0042	Failed to operate the group. Error message: %s.	Select a correct group based on the error message and try again.
Cdm.0043	Failed to trigger data extraction or loading failed. Cause: %s.	Contact customer service or technical support.
Cdm.0051	Invalid submission engine: *%s*.	Specify a correct job engine and try again.
Cdm.0052	Job *%s* is running.	The operation cannot be performed because the job is running. Try again after the job completes.
Cdm.0053	Job *%s* is not running.	Run the job and try again.
Cdm.0054	Job *%s* does not exist.	Check whether the job exists.
Cdm.0055	Unsupported job type.	Specify a correct job type and try again.
Cdm.0056	Failed to submit the job. Cause: *%s*.	Locate the cause based on the error message, rectify the fault, and try again.
Cdm.0057	Invalid job execution engine: *%s*.	Specify a correct job engine and try again.
Cdm.0058	Invalid combination of submission and execution engines.	Specify a correct job engine and try again.
Cdm.0059	Job *%s* has been disabled. Failed to submit the job.	Create a job and try again. Alternatively, contact customer service or technical support.
Cdm.0060	Link *%s* for this job has been disabled. Failed to submit the job.	Change the link and submit the job again.
Cdm.0061	Connector *%s* does not support the specified direction. Failed to submit the job.	The connector cannot be used as the source or destination of a job. Change the link and submit the job again.

Error Code	Error Message	Solution
Cdm.0062	The binary file is applicable only to the SFTP, FTP, HDFS, or OBS connector.	Specify a correct connector and try again.
Cdm.0063	An error occurred during table creation. Cause: *%s*.	Locate the cause based on the error message, rectify the fault, and try again.
Cdm.0064	The data format is incorrect.	Check whether the data format is correct based on the error message. If it is not, correct it and try again.
Cdm.0065	Failed to start the scheduler. Cause: *%s*.	Contact customer service or technical support.
Cdm.0066	Failed to obtain the sample value. Cause: *%s*.	Contact customer service or technical support.
Cdm.0067	Failed to obtain the schema. Cause: *%s*.	Contact customer service or technical support.
Cdm.0068	Failed to clear table data. Cause: %s.	• Check whether the current account has the operation permissions on the table. • Check whether the table is locked. • If neither of the preceding methods is feasible, contact customer service or technical support.
Cdm.0070	Failed to run task %s because the maximum number of running jobs has been reached.	Contact customer service or technical support.
Cdm.0071	Failed to obtain table data. Cause: %s.	Contact customer service or technical support.
Cdm.0074	Failed to repair the table. Cause: %s.	Contact customer service or technical support.
Cdm.0075	Failed to delete the table. Cause: %s.	• Check whether the current account has the operation permissions on the table. • Check whether the table is locked. • If neither of the preceding methods is feasible, contact customer service or technical support.

Error Code	Error Message	Solution
Cdm.0080	Invalid username.	Correct the username based on the error message and try again.
Cdm.0081	Invalid certificate.	Contact customer service or technical support.
Cdm.0082	The certificate is not readable.	Contact customer service or technical support.
Cdm.0083	A process cannot be configured with multiple certificates. Restart to use the new certificate.	Modify the certificate based on the error message and restart the system.
Cdm.0085	The value exceeds the upper limit.	Contact customer service or technical support.
Cdm.0088	Incorrect *XX* configuration item.	Modify the configuration item based on the error message and try again.
Cdm.0089	The configuration item XX does not exist.	• Modify the configuration item based on the error message and try again. • During the switchover from a CDM cluster of an earlier version to a CDM cluster of a later version, configuration items may be unavailable occasionally when you create a data connection or save a job. In this case, manually clear the cache and try again.
Cdm.0091	The patches cannot be installed.	Contact customer service or technical support.
Cdm.0092	The backup file does not exist.	Contact customer service or technical support.
Cdm.0093	Failed to load the krb5.conf file.	Contact customer service or technical support.
Cdm.0094	The link named *XX* does not exist.	Check whether the *XX* link exists based on the error message and try again.
Cdm.0095	The job named *XX* does not exist.	Check whether the *XX* job exists based on the error message and try again.
Cdm.0100	Job [*%s*] does not exist.	Specify a correct job and try again.

Error Code	Error Message	Solution
Cdm.0101	Link [*%s*] does not exist.	Specify a correct link and try again.
Cdm.0102	Connector [*%s*] does not exist.	Specify a correct connector and try again.
Cdm.0104	The job name exists.	Rename the job and try again.
Cdm.0105	The expression is empty.	• Check whether the expression is valid by referring to the help document. • If the fault persists, contact customer service or technical support.
Cdm.0106	Failed to calculate the *XX* expression.	• Check whether the expression is valid by referring to the help document. • If the fault persists, contact customer service or technical support.
Cdm.0107	The task is being executed. Modify job configurations later.	After the task is complete, modify the job configurations.
Cdm.0108	Failed to query table records.	• Ensure that the custom SQL statement is correct. • Ensure that the query does not time out (less than 60s). • If the preceding errors cannot be avoided, contact customer service or technical support.
Cdm.0109	The length of a job or link name cannot exceed %s.	Modify the job or link name based on the error message.
Cdm.0110	Invalid name. The name must start with a character or digit and consist of only letters, digits, underscores (_), hyphens (-), and dots (.).	Change the name based on the error message.
Cdm.0201	Failed to obtain the instance.	Contact customer service or technical support.
Cdm.0202	Unknown job status.	Try again later or contact customer service or technical support.
Cdm.0204	No MRS link is created.	Go to the **Link Management** page to create an MRS link and try again.

Error Code	Error Message	Solution
Cdm.0230	Failed to load the specified class: *%s*.	Contact customer service or technical support.
Cdm.0231	Failed to initialize the specified class: *%s*.	Contact customer service or technical support.
Cdm.0232	Failed to write data. Cause: *%s*.	Contact customer service or technical support.
Cdm.0233	An exception occurred during data extraction. Cause: *%s*.	Contact customer service or technical support.
Cdm.0234	An exception occurred during data loading. Cause: *%s*.	Contact customer service or technical support.
Cdm.0235	All data has been consumed. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0236	Invalid partitions have been retrieved from Partitioner.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0237	Failed to find the JAR file of the connector.	Contact customer service or technical support.
Cdm.0238	*%s* cannot be empty.	Modify the parameter settings based on the error message and try again.
Cdm.0239	Failed to obtain HDFS. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0240	Failed to obtain the status of the *%s* file.	Contact customer service or technical support.
Cdm.0241	Failed to obtain the type of the *%s* file.	Contact customer service or technical support.
Cdm.0242	An exception occurred during file check: *%s*.	Contact customer service or technical support.
Cdm.0243	Failed to rename *%s* to *%s*.	Rename the job and try again.
Cdm.0244	Failed to create the *%s* file.	Check whether you have the permissions or try again later. If the fault persists, contact customer service or technical support.

Error Code	Error Message	Solution
Cdm.0245	Failed to delete the *%s* file.	Check whether you have the permissions or try again later. If the fault persists, contact customer service or technical support.
Cdm.0246	Failed to create the *%s* directory.	Check whether you have the permissions or try again later. If the fault persists, contact customer service or technical support.
Cdm.0247	HBase operation failure. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0248	Failed to clear data in *%s*. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0249	The file name *%s* is invalid.	Modify the file name and try again.
Cdm.0250	Failed to perform operations in the path: *%s*.	Check whether you have the permissions or try again later. If the fault persists, contact customer service or technical support.
Cdm.0251	Failed to load data to HBase. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0307	Failed to release the link. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0315	The link name *%s* already exists.	Specify another link name and try again.
Cdm.0316	The link that does not exist cannot be updated.	Specify a correct link and try again.
Cdm.0317	Link *%s* is invalid.	Specify a correct link and try again.
Cdm.0318	The job exists and cannot be created repeatedly.	Specify another job name and try again.
Cdm.0319	The job that does not exist cannot be updated.	Check whether the job to be updated exists. If it does, modify the job name and try again.

Error Code	Error Message	Solution
Cdm.0320	Job *%s* is invalid.	Contact customer service or technical support.
Cdm.0321	Link *%s* has been used.	Release the link and try again.
Cdm.0322	Job *%s* has been used.	Contact customer service or technical support.
Cdm.0323	The submission already exists and cannot be created repeatedly.	Try again later.
Cdm.0327	Invalid link or job: *%s*.	Specify a correct link or job and try again.
Cdm.0411	An error occurred when connecting to the file server.	Contact customer service or technical support.
Cdm.0412	An error occurred when disconnecting from the file server.	Contact customer service or technical support.
Cdm.0413	An error occurred in data transfer to the file server.	Contact customer service or technical support.
Cdm.0415	An error occurred when downloading files from the file server.	Contact customer service or technical support.
Cdm.0416	An error occurred during data extraction.	Contact customer service or technical support.
Cdm.0420	The source file or source directory does not exist.	Check whether the source file or source directory exists. If it does not, specify a correct source file or directory and try again.
Cdm.0423	Duplicate files exist in the destination path.	Delete duplicate files from the destination path and try again.
Cdm.0500	The source directory or the [*%s*] file does not exist.	Specify a correct source file or directory and try again.
Cdm.0501	Invalid URI [*%s*].	Specify a correct URI and try again.
Cdm.0518	Failed to connect to HDFS. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.

Error Code	Error Message	Solution
Cdm.0523	Connection timed out due to insufficient user permissions.	Create another service user, grant required permissions to the user, and try again.
Cdm.0600	Failed to connect to the FTP server. Cause: **%s**.	It is possible that the network is disconnected, no security group or firewall rule is configured to allow access, the FTP host name cannot be parsed, or the FTP username or password is incorrect. If the fault persists, contact customer service or technical support.
Cdm.0700	Failed to connect to the SFTP server. Cause: **%s**.	It is possible that the network is disconnected, no security group or firewall rule is configured to allow access, the SFTP host name cannot be parsed, or the SFTP username or password is incorrect. If the fault persists, contact customer service or technical support.
Cdm.0800	Failed to connect to the OBS server. Cause: **%s**.	It is possible that the OBS endpoint is inconsistent with the current region, the AK/SK pair is incorrect, the AK/SK pair is not the current user's, or no security group or firewall rule is configured to allow access. If the fault persists, contact customer service or technical support.
Cdm.0801	OBS bucket [*%s*] does not exist.	The OBS bucket may not exist or is not in the current region. Specify a correct OBS bucket and try again.
Cdm.0900	Table [*%s*] does not exist.	Specify a correct table name and try again.
Cdm.0901	Failed to connect to the database server. Cause: *%s*.	Contact customer service or technical support.
Cdm.0902	Failed to execute the SQL statement. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.

Error Code	Error Message	Solution
Cdm.0903	Failed to obtain metadata. Cause: *%s*.	Check whether the quote character is correct or whether the database table exists when you create the link. If the fault persists, contact customer service or technical support.
Cdm.0904	An error occurred while retrieving data from the result. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0905	No partition column is found.	Specify a partition column and try again.
Cdm.0906	No boundary is found in the partition column.	Contact customer service or technical support.
Cdm.0911	The table name or SQL must be specified.	Specify a table name or SQL statement and try again.
Cdm.0912	The table name and SQL cannot be specified at the same time.	Specify one of them and try again.
Cdm.0913	Schema and SQL cannot be specified at the same time.	Specify one of them and try again.
Cdm.0914	Partition column is mandatory for query-based import.	Specify a partition column and try again.
Cdm.0915	The SQL-based import mode and **columnList** cannot be used at the same time.	Use either of them and try again.
Cdm.0916	Last value is mandatory for incremental read.	Specify the last value and try again.
Cdm.0917	Last value cannot be obtained without field check.	Contact customer service or technical support.
Cdm.0918	If no transfer table is specified, **shouldClearStageTable** cannot be specified.	Specify a transfer table and try again.
Cdm.0921	Type *%s* is not supported.	Specify a correct type and try again.
Cdm.0925	The partition column contains unsupported values.	Correct the values and try again.

Error Code	Error Message	Solution
Cdm.0926	Failed to obtain the schema. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0927	The transfer table is not empty.	Specify an empty transfer table and try again.
Cdm.0928	An error occurred when data is migrated from the transfer table to the destination table.	Contact customer service or technical support.
Cdm.0931	Schema column size [*%s*] does not match the result set column size [*%s*].	Change the schema column size to be the same as the result set column size and try again.
Cdm.0932	Failed to obtain the maximum value of the field.	Contact customer service or technical support.
Cdm.0934	Multiple tables of the same name exist in different schemas or catalogs.	Contact customer service or technical support.
Cdm.0935	No primary key. Specify the partition column.	Specify a partition column and try again.
Cdm.0936	The maximum number of error dirty data records has been reached.	Edit the job and increase the number of error dirty data records.
Cdm.0940	Failed to match the exact table name.	Specify a correct table name and try again.
Cdm.0941	Failed to connect to the server. Cause: *%s*.	Check whether the IP address, host name, and port number are correct, and whether the network security group and firewall are correctly configured. Locate the fault based on the error message. If the fault persists, contact customer service or technical support.
Cdm.0950	Failed to connect to the database with the existing authentication information.	Incorrect authentication information. Correct it and try again.
Cdm.0960	Server address must be specified.	Specify the server address and try again.
Cdm.0961	Invalid server address format.	Change to the correct format and try again.

Error Code	Error Message	Solution
Cdm.0962	The host IP address must be specified.	Specify the host IP address and try again.
Cdm.0963	The host port must be specified.	Specify the host port and try again.
Cdm.0964	The database must be specified.	Specify a database and try again.
Cdm.1000	Hive table [*%s*] does not exist.	Specify a correct Hive table name and try again.
Cdm.1010	Invalid URI [*%s*]. The URI must be either null or a valid URI.	Specify a correct URI and try again. Correct URI examples: • hdfs://example.com:8020/ • hdfs://example.com/ • file:/// • file:///tmp • file://localhost/tmp
Cdm.1011	Failed to connect to Hive. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1012	Failed to initialize the Hive client. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1100	Table [*%s*] does not exist.	Enter a correct table name and try again.
Cdm.1101	Failed to obtain the link. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1102	Failed to create the table. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1103	No rowkey is set.	Set the rowkey and try again.
Cdm.1104	Failed to open the table. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.

Error Code	Error Message	Solution
Cdm.1105	Failed to initialize the job. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1111	The table name is mandatory.	Specify a correct table name and try again.
Cdm.1112	The import mode is mandatory.	Set the import mode and try again.
Cdm.1113	Whether to clear data before import has not been specified.	Set **Clear Data Before Import** and try again.
Cdm.1114	The rowkey is empty. Set it in field mapping.	Fix the error based on the error message.
Cdm.1115	**Columns** is empty. Set it in field mapping.	Fix the error based on the error message.
Cdm.1116	Duplicate column names. Set it in field mapping.	Fix the error based on the error message.
Cdm.1117	An error occurred when checking whether the table exists. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1118	Table *%s* does not contain the *%s* column family.	Specify a column family and try again.
Cdm.1119	The number of column families is *%s* and the number of columns is *%s*.	Change the number of column families to the same as the number of columns and try again.
Cdm.1120	The table contains data. Clear the table data or set the configuration item to specify whether to clear the table data before the import.	Fix the error based on the error message.
Cdm.1121	Failed to close the link. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1201	Failed to connect to the Redis server. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.

Error Code	Error Message	Solution
Cdm.1202	Failed to connect to the Redis cluster in single-node mode.	Connect to the Redis cluster in another mode.
Cdm.1203	Failed to extract data from the Redis server. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1205	Redis Key Prefix cannot be blank.	Delete the whitespace before the Redis prefix and try again.
Cdm.1206	The storage type of the Redis value must be **STRING** or **HASH**.	Fix the error based on the error message.
Cdm.1207	When the value of the storage type is **STRING**, **Value Delimiter** must be specified.	Specify a value delimiter and try again.
Cdm.1208	**columnList** of Redis must be specified.	Specify **columnList** and try again.
Cdm.1209	Redis Key Delimiter cannot be empty.	Enter a correct delimiter and try again.
Cdm.1210	**primaryKeyList** of Redis must be specified.	Specify **primaryKeyList** and try again.
Cdm.1211	**primaryKeyList** of Redis must exist in **columnList**.	Specify **primaryKeyList** and try again.
Cdm.1212	**databaseType** of Redis must be **Original** or **DCS**.	Fix the error based on the error message.
Cdm.1213	**Redis Server Address** must be specified.	Specify **Redis Server Address** and try again.
Cdm.1301	Failed to connect to the MongoDB server. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1302	Failed to extract data from the MongoDB server. Cause: *%s*	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1304	The collection of MongoDB servers must be specified.	Specify the collection of MongoDB servers and try again.
Cdm.1305	**Server Address** of MongoDB must be specified.	Specify **Server Address** and try again.

Error Code	Error Message	Solution
Cdm.1306	The database name of the MongoDB service must be specified.	Specify a database and try again.
Cdm.1307	**serverlist** of MongoDB must be specified.	Specify **serverlist** and try again.
Cdm.1400	Failed to connect to the NAS server.	Contact customer service or technical support.
Cdm.1401	No permissions to access the NAS server.	Apply for the permissions and try again.
Cdm.1501	Failed to connect to the Elasticsearch server. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1502	Failed to write data to the Elasticsearch server. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1503	Failed to close the Elasticsearch link. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1504	An error occurred when obtaining the Elasticsearch index. Cause: *%s*	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1505	An error occurred when obtaining the Elasticsearch type. Cause: *%s*	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1506	An error occurred when obtaining the Elasticsearch field. Cause: *%s*	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1507	An error occurred when obtaining the Elasticsearch sample data. Cause: *%s*	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1508	The host name or IP address of the Elasticsearch server must be specified.	Specify the host name or IP address and try again.

Error Code	Error Message	Solution
Cdm.1509	The port of the Elasticsearch server must be specified.	Specify a port and try again.
Cdm.1510	The Elasticsearch index must be specified.	Specify an index and try again.
Cdm.1511	The Elasticsearch type must be specified.	Specify a type and try again.
Cdm.1512	**columnList** of Elasticsearch must be specified.	Specify **columnList** and try again.
Cdm.1513	**columnList** must contain the field type definition.	Include the field type definition and try again.
Cdm.1514	**columnList** must contain **primaryKey**.	Set the primary key field and try again.
Cdm.1515	An error occurred when resolving the JSON character string. Cause: *%s*.	Locate the cause based on the error message, rectify the fault, and try again. If the fault persists, contact customer service or technical support.
Cdm.1516	The column name *%s* is invalid.	Enter a correct column name and try again.
Cdm.1517	An error occurred when obtaining the number of documents.	Contact customer service or technical support.
Cdm.1518	The partition fails to be created.	Contact customer service or technical support.
Cdm.1519	An error occurred during data extraction.	Contact customer service or technical support.
Cdm.1520	Failed to obtain the type. Cause: *%s*.	Locate the cause based on the error message. If the fault persists, contact customer service or technical support.
Cdm.1601	Failed to connect to the server.	Contact customer service or technical support.
Cdm.1603	Failed to obtain the sample value of the *%s* topic.	Contact customer service or technical support.
Cdm.1604	No data exists in topic *%s*.	Locate the cause. Alternatively, change the topic and try again.
Cdm.1605	Invalid **brokerList**.	Specify a correct **brokerList** and try again.

4.9 Key Operation Guide

4.9.1 Incremental Migration

4.9.1.1 Incremental File Migration

CDM supports incremental migration of file systems. After full migration is complete, all new files or only specified directories or files can be exported.

Currently, CDM supports the following incremental migration modes:

1. **Exporting the files in a specified directory**
   - Application scenarios: The migration source is a file system (OBS/HDFS/FTP/SFTP). In incremental migration, only the specified files are written to the migration destination. The existing records are not updated or deleted.
   - Key configurations: **File/Path Filter** and Schedule Execution
   - Prerequisites: The source directory or file name contains the time field.
2. **Exporting the files modified after the specified time point**
   - Application scenarios: The migration source is a file system (OBS/HDFS/FTP/SFTP). The specified time point refers to the time when the file is modified. CDM migrates the files modified at or after the specified time point.
   - Key configurations: **Time Filter** and Schedule Execution
   - Prerequisites: None

NOTE

If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).

File/Path Filter

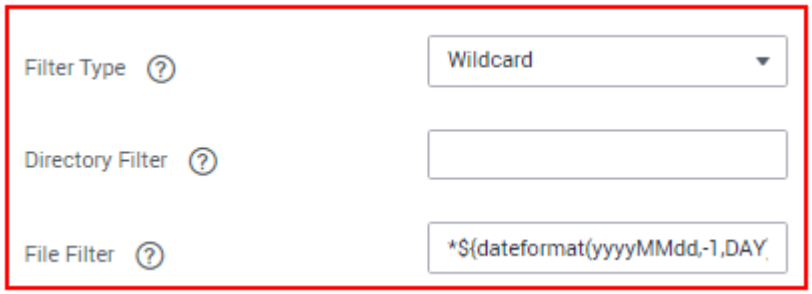
- Parameter position: When creating a table/file migration job, if the migration source is a file system, set **Filter Type** in advanced attributes of **Source Job Configuration** to **Wildcard** or **Regular expression**.
- Parameter principle: If you select **Wildcard** for **Filter Type**, CDM filters files or paths based on the configured wildcard character and migrates only files or paths that meet the specified condition.
- Example configurations:

  Suppose that the source file name contains the date and time field, such as **2017-10-15 20:25:26**, the **/opt/data/file_20171015202526.data** file is generated. Set the parameters as follows:

  a. **Filter Type**: Select **Wildcard**.

b. **File Filter**: Enter **"*${dateformat(yyyyMMdd,-1,DAY)}*"**, which is the format of the macro variables of date and time supported by CDM. For details, see **Using Macro Variables of Date and Time**.

Figure 4-53 Filtering files



c. Schedule Execution: Set **Cycle (days)** to **1**.

In this way, you can import the files generated in the previous day to the destination directory every day to implement incremental synchronization.

In incremental file migration, **Path Filter** is used in the same way as **File Filter**. The path name must contain the time field. In this case, all files in the specified path can be synchronized periodically.

Time Filter

- Parameter position: When creating a table/file migration job, if the migration source is a file system, set select **Yes** for **Time Filter**.
- Parameter principle: After you specify the start time and end time, only files that are modified between the start time (included) and end time (excluded) will be migrated.
- Example configurations:

  For example, if you want CDM to synchronize only the files generated from January 1, 2021 to January 1, 2022 to the destination, configure the following parameters:

  a. **Time Filter**: select **Yes**.

  b. **Minimum Timestamp**: Enter a value in the format of *yyyy-MM-dd HH:mm:ss*, such as **2021-01-01 00:00:00**.

  c. **Maximum Timestamp**: Enter a value in the format of *yyyy-MM-dd HH:mm:ss*, such as **2022-01-01 00:00:00**.

Figure 4-54 Time Filter

In this way, the CDM job migrates only the files generated from January 1, 2021 to January 1, 2022, and performs incremental synchronization next time it is started.

4.9.1.2 Incremental Migration of Relational Databases

CDM supports incremental migration of relational databases. After a full migration is complete, data in a specified period can be incrementally migrated. For example, data added on the previous day can be exported at 00:00:00 every day.

- **Migrating incremental data within a specified period of time**
  - Application scenarios: The source end is a relational database. The destination end can be of any type.
  - Key configurations: **WHERE Clause** and Schedule Execution
  - Prerequisites: The data table contains a date and time field or timestamp field.

In incremental migration, only the specified data is written to the data table. The existing records are not updated or deleted.

NOTE

If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).

WHERE Clause

- Parameter position: When creating a table/file migration job, if the source end is a relational database, the **Where Clause** parameter is available in the advanced attributes of **Source Job Configuration**.
- Parameter principle: Set **WHERE Clause** to an SQL statement, for example, **age > 18 and age <= 60**, CDM exports only the data that meets the SQL statement requirement. If **WHERE Clause** is not specified, the entire table is exported.

  Where Clause can be set to **macro variables of date and time**. When the data table contains the **date** or **timestamp** field, **Where Clause** and Schedule Execution can be used together to extract data of a specified date.
- Example configurations:

  Suppose that the database table contains column **DS** indicating the time, the value type of the column is **varchar(30)**, and the inserted time format is similar to *2017-xx-xx*. See **Figure 4-55**. Set the parameters as follows:

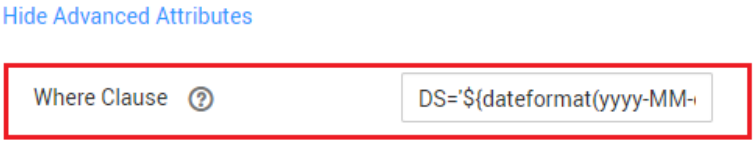
Figure 4-55 Table data

SELECT * FROM SQOOP.CDM_20171016

	FOO	BAR	DS
1	5	s	2017-05-01
2	5	s	2017-05-01
3	1	g	2017-05-02
4	4	o	2017-05-02
5	6	a	2017-05-02
6	7	n	2017-05-02
7	1	g	2017-05-02
8	4	o	2017-05-02
9	6	a	2017-05-02
10	7	n	2017-05-02
11	2	fa	2017-10-15
12	3	te	2017-10-15
13	2	fa	2017-10-15
14	3	te	2017-10-15

a. **WHERE Clause**: Set this parameter to **DS='${dateformat(yyyy-MM-dd,-1,DAY)}'**.

Figure 4-56 WHERE Clause



b. Scheduling job execution: Set **Cycle (days)** to **1** and **Start Time** to **00:00:00**.

In this way, all data generated on the previous day can be exported at 00:00:00 every day. **WHERE Clause** can be configured to various **macro variables of date and time**. You can use the macro variables of date and time and scheduled jobs with specified cycle of minutes, hours, days, weeks, or months together to automatically export data at a specific time.

4.9.1.3 HBase/CloudTable Incremental Migration

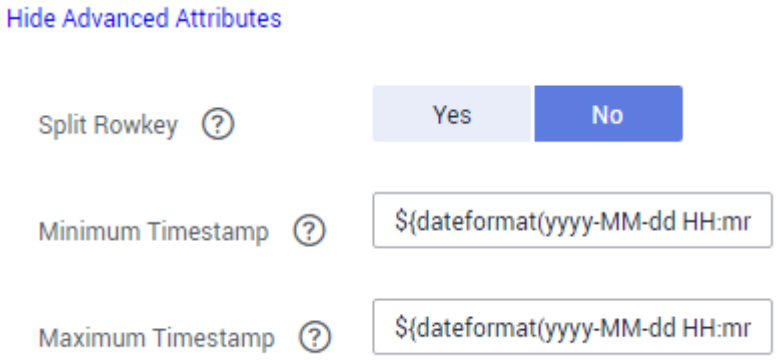
You can use CDM to export data in a specified period of time from HBase (including MRS HBase, FusionInsight HBase, and Apache HBase) and CloudTable. The CDM scheduled jobs can be used together to implement incremental migration of HBase and CloudTable.

NOTE

If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).

When creating a table/file migration job and selecting the link to HBase or CloudTable as the source link, you can set the time range in advanced attributes.

Figure 4-57 Time range



- Start time (including the value) for extracting data. The format is *yyyy-MM-dd HH:mm:ss*. Only the data generated at the specified time and later is extracted.
- End time (excluding the value) for extracting data. The format is *yyyy-MM-dd HH:mm:ss*. Only the data generated before the time point is extracted.

The two parameters can be set to **macro variables of date and time**. Examples are as follows:

- If **Minimum Timestamp** is set to **${dateformat(yyyy-MM-dd HH:mm:ss, -1, DAY)}**, only the data generated after the day before is exported.
- If **Maximum Timestamp** is set to **${dateformat(yyyy-MM-dd HH:mm:ss)}**, only the data generated before the specified time point is exported.

If both parameters are configured, CDM exports only the data generated on the previous day. In addition, if the job is configured to execute at 00:00:00 every day, the data generated every day can be incrementally synchronized.

4.9.1.4 MongoDB/DDS Incremental Migration

By using CDM, you can export MongoDB or DDS data within a specified period. With the scheduled jobs of CDM, you can implement incremental migration of MongoDB and DDS.

NOTE

If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).

When creating a table/file migration job and selecting the link to MongoDB or DDS as the source link, you can set the query filters in advanced attributes.

Figure 4-58 Setting query filters

You can set this parameter to a **macro variable of date and time**, for example, **{"ts":{$gte:ISODate("${dateformat(yyyy-MM-dd'T'HH:mm:ss.SSS'Z',-1,DAY)}")}}**, which indicates searching for the values in the **ts** field that are greater than those after time macro conversion, that is, only the data generated after the previous day is exported.

After this parameter is set, CDM exports only the data generated on the previous day. In addition, you can set the job to be executed at 00:00:00 every day, so that the data generated every day can be incrementally synchronized.

4.9.2 Using Macro Variables of Date and Time

During the creation of table/file migration jobs, CDM supports the macro variables of date and time in the following parameters of the source and destination links:

- Source directory or file
- Source table name
- Directory filter and file filter of the **wildcard** type
- Start time and end time of the **time filter** type
- Partition filter criteria and where clause
- Write directory
- Destination table name

You can use the **${}** macro variable definition identifier to define the macros of the time type. currently, dateformat and timestamp are supported.

By using the macro variables of date and time and scheduled job, you can implement incremental synchronization of databases and files.

NOTE

If you have configured a macro variable of date and time and schedule a CDM job through DataArts Studio DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).

dateformat

dateformat supports two types of parameters:

- **dateformat(format)**

  format indicates the date and time format. For details about the format definition, see the definition in **java.text.SimpleDateFormat.java**.

  For example, if the current date is **2017-10-16 09:00:00**, **yyyy-MM-dd HH:mm:ss** indicates **2017-10-16 09:00:00**.

- dateformat(format, dateOffset, dateType)
  - **format** indicates the format of the returned date.
  - **dateOffset** indicates the date offset.
  - **dateType** indicates the type of the date offset.

    Currently, **dateType** supports SECOND, MINUTE, HOUR, MONTH, YEAR, and DAY.

    NOTE

    Pay attention to the following special scenarios of **MONTH** and **YEAR**:

    - If the date does not exist after the offset, the latest date of the month in the calendar is used.
    - These two offset types cannot be used for the start time and end time in the **Time Filter** parameter of the source and destination jobs.

  For example, if the current date is **2023-03-01 09:00:00**, then:

  - **dateformat(yyyy-MM-dd HH:mm:ss, -1, YEAR)** indicates the year before the current time, that is, **2022-03-01 09:00:00**.
  - **dateformat(yyyy-MM-dd HH:mm:ss, -3, MONTH)** indicates three months before the current time, that is, **2022-12-01 09:00:00**.
  - **dateformat(yyyy-MM-dd HH:mm:ss, -1, DAY)** indicates the day before the current time, that is, **2023-02-28 09:00:00**.
  - **dateformat(yyyy-MM-dd HH:mm:ss, -1, HOUR)** indicates one hour before the current time, that is, **2023-03-01 08:00:00**.
  - **dateformat(yyyy-MM-dd HH:mm:ss, -1, MINUTE)** indicates one minute before the current time, that is, **2023-03-01 08:59:00**.
  - **dateformat(yyyy-MM-dd HH:mm:ss, -1, SECOND)** indicates one second before the current time, that is, **2023-03-01 08:59:59**.

timestamp

timestamp supports two types of parameters:

- **timestamp()**

  Indicates the returned timestamp of the current time, that is, the number of milliseconds that have elapsed since 00:00:00 on January 1, 1970 (1970-01-01 00:00:00 GMT). For example, 1508078516286.

- **timestamp(dateOffset, dateType)**

  Indicates the timestamp returned after time offset. **dateOffset** and **dateType** indicate the date offset and the offset type, respectively.

  For example, if the current date is **2017-10-16 09:00:00**, **timestamp(-10, MINUTE)** indicates that the timestamp generated 10 minutes before the current time point is returned, that is, **1508115000000**.

Macro Variable Definition of Time and Date

Suppose that the current time is **2017-10-16 09:00:00**, then **Table 4-106** describes the macro variable definitions of time and date.

Table 4-106 Macro variable definition of time and date

Macro Variable	Description	Display Effect
${dateformat(yyyy-MM-dd)}	Returns the current date in **yyyy-MM-dd** format.	2017-10-16
${dateformat(yyyy/MM/dd)}	Returns the current date in **yyyy/MM/dd** format.	2017/10/16
${dateformat(yyyy_MM_dd HH:mm:ss)}	Returns the current time in **yyyy_MM_dd HH:mm:ss** format.	2017_10_16 09:00:00
${dateformat(yyyy-MM-dd HH:mm:ss, -1, DAY)}	Returns the current time in **yyyy-MM-dd HH:mm:ss** format. The date is one day before the current day.	2017-10-15 09:00:00
${timestamp()}	Returns the timestamp of the current time, that is, the number of milliseconds that have elapsed since 00:00:00 on January 1, 1970.	1508115600000
${timestamp(-10, MINUTE)}	Returns the timestamp generated 10 minutes before the current time point.	1508115000000
${timestamp(dateformat(yyyyMMdd))}	Returns the timestamp of 00:00:00 of the current day.	1508083200000
${timestamp(dateformat(yyyyMMdd,-1,DAY))}	Returns the timestamp of 00:00:00 of the previous day.	1507996800000
${timestamp(dateformat(yyyyMMddHH))}	Returns the timestamp of the current hour.	1508115600000

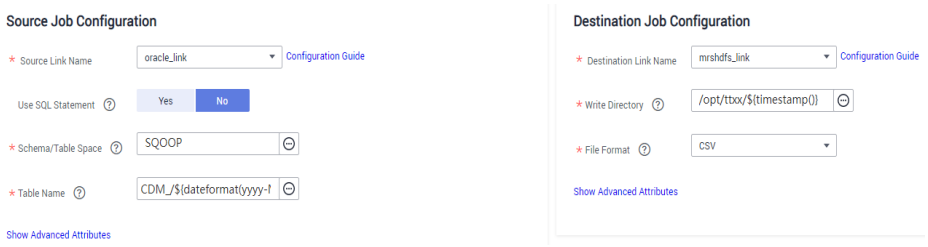
Time and Date Macro Variables of Paths and Table Names

Figure 4-59 shows an example. If:

- **Table Name** under **Source Link Configuration** is set to **CDM_/${dateformat(yyyy-MM-dd)}**.
- **Write Directory** under **Destination Link Configuration** is set to **/opt/ttxx/${timestamp()}**.

After the macro definition conversion, this job indicates that data in table **SQOOP.CDM_20171016** in the Oracle database is migrated to the **/opt/ttxx/1508115701746** directory of the HDFS server.

Figure 4-59 Setting **Table Name** and **Write Directory** to a time and date macro variable



Currently, a table name or path name can contain multiple macro variables. For example, **/opt/ttxx/${dateformat(yyyy-MM-dd)}/${timestamp()}** is converted to **/opt/ttxx/2017-10-16/1508115701746**.

Time and Date Macro Variables in the Where Clause

Figure 4-60 uses table **SQOOP.CDM_20171016** as an example. The table contains column **DS**, which indicates the time.

Figure 4-60 Table data

SELECT * FROM SQOOP.CDM_20171016

	FOO	BAR	DS
1	5	snap	2017-05-01
2	5	snap	2017-05-01
3	1	google	2017-05-02
4	4	oracle	2017-05-02
5	6	amd	2017-05-02
6	7	nvda	2017-05-02
7	1	google	2017-05-02
8	4	oracle	2017-05-02
9	6	amd	2017-05-02
10	7	nvda	2017-05-02
11	2	facebook	2017-10-15
12	3	tesla	2017-10-15
13	2	facebook	2017-10-15
14	3	tesla	2017-10-15

Suppose that the current date is **2017-10-16** and you want to export data generated the day before the current day (DS = 2017-10-15), then you can set the value of **Where Clause** to **DS='${dateformat(yyyy-MM-dd,-1,DAY)}'** when creating a job. In this way, you can export all data that complies with the DS = 2017-10-15 condition.

Implementing Incremental Synchronization by Configuring the Macro Variables of Date and Time and Scheduled Jobs

Two simple application scenarios are as follows:

- The database table contains column **DS** that indicates the time, the value type of the column is **varchar(30)**, and the inserted time format is similar to **2017-xx-xx**.

  In a scheduled job, the cycle is one day, and the scheduled job is executed at 00:00:00 every day. Set the value of **Where Clause** to **DS='${dateformat(yyyy-MM-dd,-1,DAY)}'**, and then data generated in the previous day will be exported at 00:00:00 every day.

- The database table contains column **time** that indicates the time, the type is **Number**, and the inserted time format is timestamp.

  In a scheduled job, the cycle is one day, and the scheduled job is executed at 00:00:00 every day. Set the value of **Where Clause** to **time between ${timestamp(-1,DAY)} and ${timestamp()}**, and then data generated on the previous day will be exported at 00:00:00 every day.

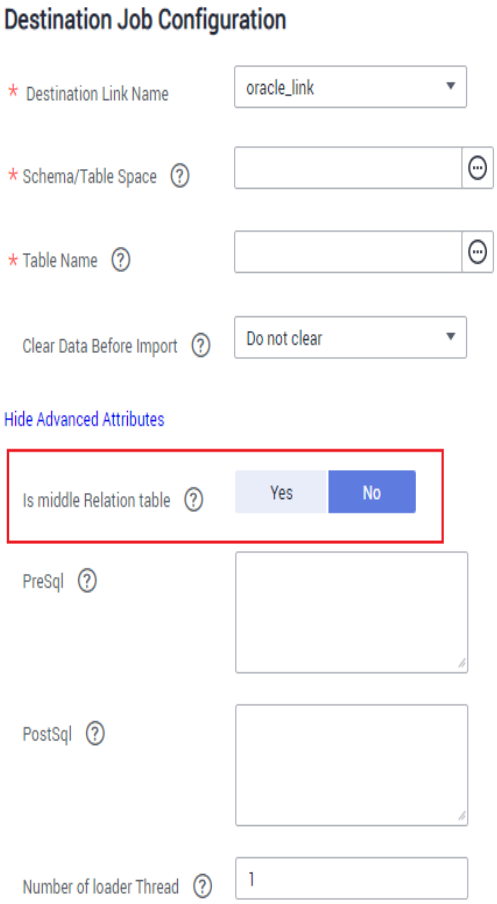
Configuration principles of other application scenarios are the same.

4.9.3 Migration in Transaction Mode

When a CDM job fails to be executed, CDM rolls back the data to the state before the job starts and automatically deletes data from the destination table.

- Parameter position: When creating a table/file migration job, if the migration source is a relational database, set **Import to Staging Table** in the advanced attributes of **Destination Job Configuration** to determine whether to enable the transaction mode.
- Parameter principle: If you set this parameter to **Yes**, CDM automatically creates a temporary table and imports the data to the temporary table. After the data is imported successfully, CDM migrates the data to the destination table in transaction mode of the database. If the import fails, the destination table is rolled back to the state before the job starts.

Figure 4-61 Migration in transaction mode



NOTE

If you select **Clear part of data** or **Clear all data** for **Clear Data Before Import**, CDM does not roll back the deleted data in transaction mode.

4.9.4 Encryption and Decryption During File Migration

When you migrate files to a file system, CDM can encrypt and decrypt those files. Currently, CDM supports the following encryption modes:

- **AES-256-GCM**
- **KMS Encryption**

AES-256-GCM

Currently, only AES-256-GCM (NoPadding) is supported. This algorithm is used for encryption at the migration destination and decryption at the migration source. The supported source and destination data sources are as follows:

- Data sources supported by the migration source: HDFS (supported in the binary format)
- Data sources supported by the migration destination: HDFS (supported in the binary format)

The following part describes how to use AES-256-GCM to decrypt the encrypted files to be exported from HDFS and encrypt the files to be imported to HDFS.

- **Configure decryption at the migration source.**

  When you use CDM to create a job for exporting files from HDFS, set the migration source to HDFS and file format to binary, and set the following parameters in the advanced settings of **Source Job Configuration**:

  a. **Encryption**: Select **AES-256-GCM**.

  b. **DEK**: The key must be the same as that configured in encryption. Otherwise, the decrypted data is incorrect and the system does not display an error message.

  c. **IV**: The initialization vector must be the same as that configured in encryption. Otherwise, the decrypted data is incorrect and the system does not display an error message.

  In this way, after CDM exports encrypted files from HDFS, the files written to the migration destination are decrypted plaintext files.

- **Configure encryption at the migration destination.**

  When you create a CDM job to import files to HDFS, set the migration destination to HDFS and file format to binary, and set the following parameters in the advanced settings of **Destination Job Configuration**:

  a. **Encryption**: Select **AES-256-GCM**.

  b. **DEK**: custom encryption key. The key consists of 64 hexadecimal numbers. It is case-insensitive but must contain 64 characters. For example, **DD0AE00DFECD78BF051BCFDA25BD4E320DB0A7AC75A1F3FC3D3C56A457DCDC1B**.

  c. **IV**: custom initialization vector. The initialization vector consists of 32 hexadecimal numbers. It is case-insensitive but must contain 32 characters. For example, **5C91687BA886EDCD12ACBC3FF19A3C3F**.

  In this way, after CDM imports files to HDFS, the files in the destination HDFS are encrypted using the AES-256-GCM algorithm.

KMS Encryption

NOTE

The migration source does not support KMS encryption.

CDM supports KMS encryption if tables, files, or a whole database is migrated to OBS. In the **Advanced Attributes** area of the **Destination Job Configuration** page, set the parameters.

A key must be created in KMS of DEW in advance. For details, see the *Data Encryption Workshop User Guide*.

After KMS encryption is enabled, objects to be uploaded will be encrypted and stored on OBS. When you download the encrypted objects, the encrypted data will be decrypted on the server and displayed in plaintext to users.

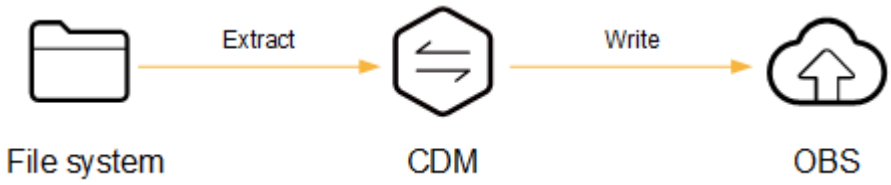
NOTE

- If KMS encryption is enabled, **MD5 verification** cannot be used.
- If the KMS ID of another project is used, change **Project ID** to the ID of the project to which KMS belongs. If KMS and CDM are in the same project, retain the default value of **Project ID**.
- After KMS encryption is performed, the encryption status of the objects on OBS cannot be changed.
- A key in use cannot be deleted. Otherwise, the object encrypted with this key cannot be downloaded.

4.9.5 MD5 Verification

CDM extracts data from the migration source and writes the data to the migration destination. **Figure 4-62** shows the migration mode when files are migrated to OBS.

Figure 4-62 Migrating files to OBS



During the process, CDM uses MD5 to verify file consistency.

- **Extract**
  - The migration source can be OBS, HDFS, FTP, SFTP, or HTTP. It can check whether the files extracted by CDM are consistent with source files.
  - This function is controlled by the **MD5 File Extension** parameter (available when **File Format** is set to **Binary**) in **Source Job Configuration**. Set this parameter to the file name extension of the MD5 file in the source file system.
  - If a source file **build.sh** and a file for saving MD5 value **build.sh.md5** are located in the same directory, and **MD5 File Extension** is configured, only the file **build.sh.md5** is migrated to the destination. Files without the MD5 value or whose MD5 values do not match fail to be migrated, and the MD5 file is not migrated.
  - If **MD5 File Extension** is not configured, all files are migrated.
- **Write**
  - Currently, this function can be used only when OBS serves as the migration destination. It can check whether the files written to OBS are consistent with those extracted from CDM.
  - This function is controlled by the **Validate MD5 Value** parameter in **Destination Job Configuration**. After the files are read and written to OBS, the MD5 value in the HTTP header is used to verify the files on OBS and the verification result is written to an OBS bucket (the bucket can be the one that does not store migration files). If the migration source does not have the MD5 file, the verification will not be performed.

NOTE

- When files are migrated to a file system, only the extracted files are verified.
- When files are migrated to OBS, both the extracted files and files written to OBS are verified.
- If MD5 verification is used, **KMS encryption** cannot be used.

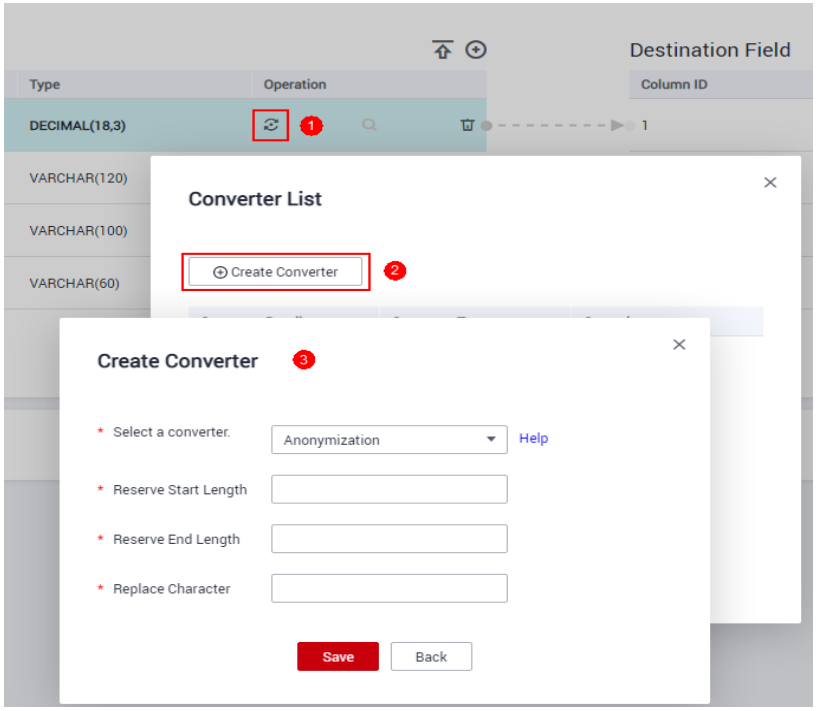
4.9.6 Configuring Field Converters

Scenario

- After the job parameters are configured, field mapping needs to be configured. You can click in the **Operation** column to create a field converter.
- If files are migrated between FTP, SFTP, OBS, and HDFS and the migration source's **File Format** is set to **Binary**, files will be directly transferred, free from field mapping.

You can create a field converter on the **Map Field** page when creating a table/file migration job.

Figure 4-63 Creating a field converter



CDM can convert fields during migration. Currently, the following field converters are supported:

- **Anonymization**
- **Trim**
- **Reverse String**
- **Replace String**

- **Remove line break**
- **Expression Conversion**

Constraints

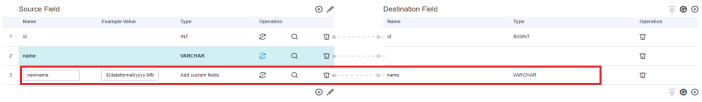
- If **Use SQL Statement** is set to **Yes** in the source job configuration, converters cannot be created.
- On the **Map Field** tab page, if CDM fails to obtain all columns by obtaining sample values (for example, when data is exported from HBase, CloudTable, or MongoDB, there is a high probability that CDM failed to obtain all columns), you can click  and select **Add a new field** to add new fields to ensure that the data imported to the migration destination is complete.
- When a relational database, Hive, DLI, or MRS Hudi is used as the migration source, sample values cannot be obtained.
- When SQLServer is the destination, fields of the timestamp type cannot be written. You must change their type (for example, to datetime) so that they can be written.
- Column names are displayed when the source of the migration job is OBS, CSV files are to be migrated, and parameter **Extract first row as columns** is set to **Yes**.
- Field converters configuration is not involved when the binary format is used to migrate files to files.
- In the automatic table creation scenario, you need to manually add fields to the destination table in advance and then add fields to the field mapping.
- After a field is added, its sample value is not displayed on the console. This does not affect the field value transmission. CDM directly writes the field value to the destination end.
- If the field mapping is incorrect, you can adjust the field mapping by dragging fields or clicking  to map fields in batches.
- An expression processes the data of a field. When you create an expression converter, do not use a time macro. If you need to use a time macro, use either of the following methods (if the source is of the file type, only **Method 1** is supported):
  - Method 1: When creating an expression converter, use two single quotation marks ('') to enclose the expression.

    For example, if expression **${dateformat(yyyy-MM-dd)}** is not enclosed in quotation marks, the hyphen (-) in the value **2017-10-16** parsed from the expression will be recognized as a minus sign, and further calculation will be performed to generate result **1991**, which is incorrect. If you enclose the expression in quotation marks, that is, **'${dateformat(yyyy-MM-dd)}'**, you will obtain **'2017-10-16'**, which is correct.

Figure 4-64 Using two single quotation marks ('') to enclose an expression

- Method 2: Add a custom source field, enter a macro variable of date and time for **Example Value**, and map the field to a destination field again.

  Figure 4-65 Adding a custom source field

- If the data is imported to GaussDB(DWS), you need to select the distribution columns in the destination fields. You are advised to select the distribution columns according to the following rules:

  a. Use the primary key as the distribution column.

  b. If multiple data segments are combined as primary keys, specify all primary keys as the distribution column.

  c. In the scenario where no primary key is available, if no distribution column is selected, DWS uses the first column as the distribution column by default. As a result, data skew risks exist.

Anonymization

This converter is used to hide key information about the character string. For example, if you want to convert **12345678910** to **123****8910**, configure the parameters as follows:

- Set **Reserve Start Length** to **3**.
- Set **Reserve End Length** to **4**.
- Set **Replace Character** to *****.

Trim

This converter is used to automatically delete the spaces before and after a string. No parameters need to be configured.

Reverse String

This converter is used to automatically reverse a string. For example, reverse **ABC** into **CBA**. No parameters need to be configured.

Replace String

This converter is used to replace a character string. You need to configure the object to be replaced and the new value.

Remove line break

This converter is used to delete the newline characters, such as \n, \r, and \r\n from the field.

Expression Conversion

This converter uses the JSP expression language (EL) to convert the current field or a row of data. The JSP EL is used to create arithmetic and logical expressions. In an expression, you can use integers, floating point numbers, strings, constants **true** and **false**, and **null**.

During data conversion, if the content to be replaced contains a special character, use a backslash (\) to escape the special character to a common one.

- The expression supports the following environment variables:
  - **value**: indicates the current field value.
  - **row**: indicates the current row, which is an array type.
- The expression supports the following Utils:
  a. If the field is of the string type, convert all character strings into lowercase letters, for example, convert **aBC** to **abc**.

     Expression: StringUtils.lowerCase(value)

  b. Convert all character strings of the current field to uppercase letters.

     Expression: StringUtils.upperCase(value)

  c. Convert the format of the first date field from 2018-01-05 15:15:05 to 20180105.

     Expression: DateUtils.format(DateUtils.parseDate(row[0],"yyyy-MM-dd HH:mm:ss"),"yyyyMMdd")

  d. Convert a timestamp to a date string in *yyyy-MM-dd hh:mm:ss* format, for example, convert **1701312046588** to **2023-11-30 10:40:46**.

     Expression: DateUtils.format(NumberUtils.toLong(value),"yyyy-MM-dd HH:mm:ss")

  e. Convert a date string in the yyyy-MM-dd hh:mm:ss format to a timestamp.

     Expression: DateUtils.getTime(DateUtils.parseDate(value,"yyyy-MM-dd hh:mm:ss"))

  f. If the field value is a date string in *yyyy-MM-dd* format, extract the year from the field value, for example, extract **2017** from **2017-12-01**.

     Expression: StringUtils.substringBefore(value,"-")

g. If the field value is of the numeric type, convert the value to a new value which is two times greater than the original value:

   Expression: value*2

h. Convert the field value **true** to **Y** and other field values to **N**.

   Expression: value=="true"?"Y":"N"

i. If the field value is of the string type and is left empty, convert it to **Default**. Otherwise, the field value will not be converted.

   Expression: empty value? "Default":value

j. Convert date format **2018/01/05 15:15:05** to **2018-01-05 15:15:05**:

   Expression: DateUtils.format(DateUtils.parseDate(value,"yyyy/MM/dd HH:mm:ss"),"yyyy-MM-dd HH:mm:ss")

k. Obtain a 36-bit universally unique identifier (UUID):

   Expression: CommonUtils.randomUUID()

l. If the field is of the string type, capitalize the first letter, for example, convert **cat** to **Cat**.

   Expression: StringUtils.capitalize(value)

m. If the field is of the string type, convert the first letter to a lowercase letter, for example, convert **Cat** to **cat**.

   Expression: StringUtils.uncapitalize(value)

n. If the field is of the string type, use a space to fill in the character string to the specified length and center the character string. If the length of the character string is not shorter than the specified length, do not convert the character string. For example, convert **ab** to meet the specified length 4.

   Expression: StringUtils.center(value,*4*)

o. Delete a newline (including **\n**, **\r**, and **\r\n**) at the end of a character string. For example, convert **abc\r\n\r\n** to **abc\r\n**.

   Expression: StringUtils.chomp(value)

p. If the string contains the specified string, **true** is returned; otherwise, **false** is returned. For example, **abc** contains **a** so that **true** is returned.

   Expression: StringUtils.contains(value,"*a*")

q. If the string contains any character of the specified string, **true** is returned; otherwise, **false** is returned. For example, **zzabyycdxx** contains either **z** or **a** so that **true** is returned.

   Expression: StringUtils.containsAny(value,"*za*")

r. If the string does not contain any one of the specified characters, **true** is returned. If any specified character is contained, **false** is returned. For example, **abz** contains one character of **xyz** so that **false** is returned.

   Expression: StringUtils.containsNone(value,"*xyz*")

s. If the string contains only the specified characters, **true** is returned. If any other character is contained, **false** is returned. For example, **abab** contains only characters among **abc** so that **true** is returned.

   Expression: StringUtils.containsOnly(value,"*abc*")

t. If the character string is empty or null, convert it to the specified character string. Otherwise, do not convert the character string. For example, convert the empty character string to null.

Expression: StringUtils.defaultIfEmpty(value,*null*)

u. If the string ends with the specified suffix (case sensitive), **true** is returned; otherwise, **false** is returned. For example, if the suffix of **abcdef** is not null, **false** is returned.

Expression: StringUtils.endsWith(value,*null*)

v. If the string is the same as the specified string (case sensitive), **true** is returned; otherwise, **false** is returned. For example, after strings **abc** and **ABC** are compared, **false** is returned.

Expression: StringUtils.equals(value,"*ABC*")

w. Obtain the first index of the specified character string in a character string. If no index is found, **-1** is returned. For example, the first index of **ab** in **aabaabaa** is 1.

Expression: StringUtils.indexOf(value,"*ab*")

x. Obtain the last index of the specified character string in a character string. If no index is found, **-1** is returned. For example, the last index of **k** in **aFkyk** is 4.

Expression: StringUtils.lastIndexOf(value,"*k*")

y. Obtain the first index of the specified character string from the position specified in the character string. If no index is found, **-1** is returned. For example, the first index of **b** obtained after the index 3 of **aabaabaa** is 5.

Expression: StringUtils.indexOf(value,"*b*",*3*)

z. Obtain the first index of any specified character in a character string. If no index is found, **-1** is returned. For example, the first index of **z** or **a** in **zzabyycdxx.** is 0.

Expression: StringUtils.indexOfAny(value,"*za*")

aa. If the string contains any Unicode character, **true** is returned; otherwise, **false** is returned. For example, **ab2c** contains only non-Unicode characters so that **false** is returned.

Expression: StringUtils.isAlpha(value)

ab. If the string contains only Unicode characters and digits, **true** is returned; otherwise, **false** is returned. For example, **ab2c** contains only Unicode characters and digits, so that **true** is returned.

Expression: StringUtils.isAlphanumeric(value)

ac. If the string contains only Unicode characters, digits, and spaces, **true** is returned; otherwise, **false** is returned. For example, **ab2c** contains only Unicode characters and digits, so that **true** is returned.

Expression: StringUtils.isAlphanumericSpace(value)

ad. If the string contains only Unicode characters and spaces, **true** is returned; otherwise, **false** is returned. For example, **ab2c** contains Unicode characters and digits so that **false** is returned.

Expression: StringUtils.isAlphaSpace(value)

ae. If the string contains only printable ASCII characters, **true** is returned; otherwise, **false** is returned. For example, for **!ab-c~**, **true** is returned.

Expression: StringUtils.isAsciiPrintable(value)

af. If the string is empty or null, **true** is returned; otherwise, **false** is returned.

Expression: StringUtils.isEmpty(value)

ag. If the string contains only Unicode digits, **true** is returned; otherwise, **false** is returned.

Expression: StringUtils.isNumeric(value)

ah. Obtain the leftmost characters of the specified length. For example, obtain the leftmost two characters **ab** from **abc**.

Expression: StringUtils.left(value,*2*)

ai. Obtain the rightmost characters of the specified length. For example, obtain the rightmost two characters **bc** from **abc**.

Expression: StringUtils.right(value,*2*)

aj. Concatenate the specified character string to the left of the current character string and specify the length of the concatenated character string. If the length of the current character string is not shorter than the specified length, the character string will not be converted. For example, if **yz** is concatenated to the left of **bat** and the length must be 8 after concatenation, the character string is **yzyzybat** after conversion.

Expression: StringUtils.leftPad(value,*8*,"*yz*")

ak. Concatenate the specified character string to the right of the current character string and specify the length of the concatenated character string. If the length of the current character string is not shorter than the specified length, the character string will not be converted. For example, if **yz** is concatenated to the right of **bat** and the length must be 8 after concatenation, the character string is **batyzyzy** after conversion.

Expression: StringUtils.rightPad(value,*8*,"*yz*")

al. If the field is of the string type, obtain the length of the current character string. If the character string is null, **0** is returned.

Expression: StringUtils.length(value)

am. If the field is of the string type, delete all the specified character strings from it. For example, delete **ue** from **queued** to obtain **qd**.

Expression: StringUtils.remove(value,"*ue*")

an. If the field is of the string type, remove the substring at the end of the field. If the specified substring is not at the end of the field, no conversion is performed. For example, remove **.com** at the end of **www.domain.com**.

Expression: StringUtils.removeEnd(value,"*.com*")

ao. If the field is of the string type, delete the substring at the beginning of the field. If the specified substring is not at the beginning of the field, no conversion is performed. For example, delete **www.** at the beginning of **www.domain.com**.

Expression: StringUtils.removeStart(value,"*www.*")

ap. If the field is of the string type, replace all the specified character strings in the field. For example, replace **a** in **aba** with **z** to obtain **zbz**.

Expression: StringUtils.replace(value,"*a*","*z*")

If the content to be replaced contains a special character, the special character must be escaped to a common character. For example, if you want to delete **\t** from a string, use the following expression:

StringUtils.replace(value,"\\t",""), which means escaping the backslash (\) again.

aq. If the field is of the string type, replace multiple characters in the character string at a time. For example, replace **h** in **hello** with **j** and **o** with **y** to obtain **jelly**.

Expression: StringUtils.replaceChars(value,"*ho*","*jy*")

ar. If the string starts with the specified prefix (case sensitive), **true** is returned; otherwise, **false** is returned. For example, **abcdef** starts with **abc**, so that **true** is returned.

Expression: StringUtils.startsWith(value,"*abc*")

as. If the field is of the string type, delete all the specified characters at the beginning and end of the field. the field. For example, delete all **x**, **y**, **z**, and **b** from **abcyx** to obtain **abc**.

Expression: StringUtils.strip(value,"*xyz*b")

at. If the field is of the string type, delete all the specified characters at the end of the field, for example, delete the "abc" string at the end of the field.

Expression: StringUtils.stripEnd(value,*"abc"*)

au. If the field is of the string type, delete all the specified characters at the beginning of the field, for example, delete all spaces at the beginning of the field.

Expression: StringUtils.stripStart(value,*null*)

av. If the field is of the string type, obtain the substring after the specified position (the index starts from 0, including the character at the specified position) of the character string. If the specified position is a negative number, calculate the position in the descending order. The first digit at the end is -1. For example, obtain the character whose index is 2 from **abcde** (that is, **c**) and the string after it, that is, **cde**.

Expression: StringUtils.substring(value,*2*)

aw. If the field is of the string type, obtain the substring in a specified range (the index starts from 0, including the character at the start and excluding the character at the end). If the range is a negative number, calculate the position in the descending order. The first digit at the end is -1. For example, obtain the string between the second character (c) and fourth character (e) of **abcde**, that is, **cd**.

Expression: StringUtils.substring(value,*2*,4)

ax. If the field is of the string type, obtain the substring after the first specified character. For example, obtain the substring after the first **b** in **abcba**, that is, **cba**.

Expression: StringUtils.substringAfter(value,"*b*")

ay. If the field is of the string type, obtain the substring after the last specified character. For example, obtain the substring after the last **b** in **abcba**, that is, **a**.

Expression: StringUtils.substringAfterLast(value,"*b*")

az. If the field is of the string type, obtain the substring before the first specified character. For example, obtain the substring before the first **b** in **abcba**, that is, **a**.

Expression: StringUtils.substringBefore(value,"*b*")

ba. If the field is of the string type, obtain the substring before the last specified character. For example, obtain the substring before the last **b** in **abcba**, that is, **abc**.

Expression: StringUtils.substringBeforeLast(value,"*b*")

bb. If the field is of the string type, obtain the substring nested within the specified string. If no substring is found, **null** is returned. For example, obtain the substring between **tag** in **tagabctag**, that is, **abc**.

Expression: StringUtils.substringBetween(value,"*tag*")

bc. If the field is of the string type, delete the control characters (char≤32) at both ends of the character string, for example, delete the spaces at both ends of the character string.

Expression: StringUtils.trim(value)

bd. Convert the character string to a value of the byte type. If the conversion fails, **0** is returned.

Expression: NumberUtils.toByte(value)

be. Convert the character string to a value of the byte type. If the conversion fails, the specified value, for example, **1**, is returned.

Expression: NumberUtils.toByte(value,*1*)

bf. Convert the character string to a value of the double type. If the conversion fails, **0.0d** is returned.

Expression: NumberUtils.toDouble(value)

bg. Convert the character string to a value of the double type. If the conversion fails, the specified value, for example, **1.1d**, is returned.

Expression: NumberUtils.toDouble(value,*1.1d*)

bh. Convert the character string to a value of the float type. If the conversion fails, **0.0f** is returned.

Expression: NumberUtils.toFloat(value)

bi. Convert the character string to a value of the float type. If the conversion fails, the specified value, for example, **1.1f**, is returned.

Expression: NumberUtils.toFloat(value,*1.1f*)

bj. Convert the character string to a value of the int type. If the conversion fails, **0** is returned.

Expression: NumberUtils.toInt(value)

bk. Convert the character string to a value of the int type. If the conversion fails, the specified value, for example, **1**, is returned.

Expression: NumberUtils.toInt(value,*1*)

bl. Convert the character string to a value of the long type. If the conversion fails, **0** is returned.

Expression: NumberUtils.toLong(value)

bm. Convert the character string to a value of the long type. If the conversion fails, the specified value, for example, **1L**, is returned.

Expression: NumberUtils.toLong(value,*1L*)

bn. Convert the character string to a value of the short type. If the conversion fails, **0** is returned.

Expression: NumberUtils.toShort(value)

bo. Convert the character string to a value of the short type. If the conversion fails, the specified value, for example, **1**, is returned.

Expression: NumberUtils.toShort(value,*1*)

bp. Convert the IP string to a value of the long type, for example, convert **10.78.124.0** to **172915712**.

Expression: CommonUtils.ipToLong(value)

bq. Read an IP address and physical address mapping file from the network, and download the mapping file to the map collection. *url* indicates the address for storing the IP mapping file, for example, **http://10.114.205.45:21203/sqoop/IpList.csv**.

Expression: HttpsUtils.downloadMap("*url*")

br. Cache the IP address and physical address mappings and specify a key for retrieval, for example, **ipList**.

Expression: CommonUtils.setCache("*ipList*",HttpsUtils.downloadMap("*url*"))

bs. Obtain the cached IP address and physical address mappings.

Expression: CommonUtils.getCache("*ipList*")

bt. Check whether the IP address and physical address mappings are cached.

Expression: CommonUtils.cacheExists("*ipList*")

bu. Based on the specified offset type (month/day/hour/minute/second) and offset (positive number indicates increase and negative number indicates decrease), convert the time in the specified format to a new time, for example, add 8 hours to **2019-05-21 12:00:00**.

Expression: DateUtils.getCurrentTimeByZone("*yyyy-MM-dd HH:mm:ss*",value, "*hour*", *8*)

bv. If the value is empty or null, "aaa" is returned. Otherwise, **value** is returned.

Expression: StringUtils.defaultIfEmpty(value,*"aaa"*)

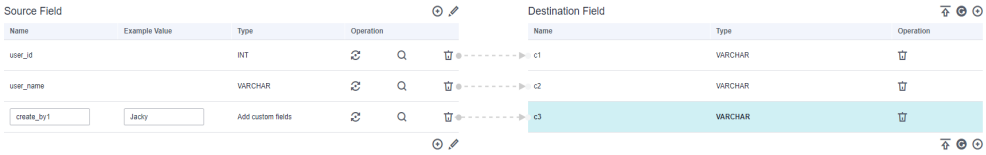
4.9.7 Adding Fields

Scenario

- After job parameters are configured, field mapping needs to be configured. You can customize new fields by clicking on the **Map Field** page.
- If files are migrated between FTP, SFTP, OBS, and HDFS and the migration source's **File Format** is set to **Binary**, files will be directly transferred, free from field mapping.
- In other scenarios, CDM automatically maps fields of the source table and the destination table. You need to check whether the mapping and time format are correct. For example, check whether the source field type can be converted into the destination field type.

You can click on the **Map Field** page and select **Add** to customize a new field. This field is usually used to mark the database source to ensure the integrity of the data imported to the migration destination.

Figure 4-66 Field mapping



Currently, the following field types are supported:

- **Constant Parameter**

  Constant parameters are fixed parameters and do not need to be reconfigured. For example, **lable** = **friends** is used to identify a constant value.

- **Variables**

  You can use variables such as time macros, table name macros, and version macros to mark database source information. The variable syntax is ${variable}, where **variable** indicates a variable. For example, **input_time** = **${timestamp()}** indicates the timestamp of the current time.

- Expression

  You can use the expression language to dynamically generate parameter values based on the running environment. The expression syntax is #{expr}, where **expr** indicates an expression. For example, **time** = **#{DateUtil.now()}** is used to identify the current date string.

Constraints

- On the **Map Field** tab page, if CDM fails to obtain all columns by obtaining sample values (for example, when data is exported from HBase, CloudTable, or MongoDB, there is a high probability that CDM failed to obtain all columns), you can click and select **Add a new field** to add new fields to ensure that the data imported to the migration destination is complete.
- When a relational database, Hive, DLI, or MRS Hudi is used as the migration source, sample values cannot be obtained.
- When SQLServer is the destination, fields of the timestamp type cannot be written. You must change their type (for example, to datetime) so that they can be written.
- Column names are displayed when the source of the migration job is OBS, CSV files are to be migrated, and parameter **Extract first row as columns** is set to **Yes**.
- Field mapping is not involved when the binary format is used to migrate files to files.
- In the automatic table creation scenario, you need to manually add fields to the destination table in advance and then add fields to the field mapping.
- After a field is added, its sample value is not displayed on the console. This does not affect the field value transmission. CDM directly writes the field value to the destination end.
- If the field mapping is incorrect, you can adjust the field mapping by dragging fields or clicking to map fields in batches.

- If the data is imported to DWS, you need to select the distribution columns in the destination fields. You are advised to select the distribution columns according to the following principles:
  a. Use the primary key as the distribution column.
  b. If multiple data segments are combined as primary keys, specify all primary keys as the distribution column.
  c. In the scenario where no primary key is available, if no distribution column is selected, DWS uses the first column as the distribution column by default. As a result, data skew risks exist.
- If a source field type is not supported, convert the field type to a type supported by CDM by referring to **Converting Unsupported Data Types**.

4.9.8 Migrating Files with Specified Names

You can migrate files (a maximum of 50) with specified names from FTP, OBS, or SFTP at a time. The exported files can only be written to the same directory on the migration destination.

When creating a table/file migration job, if the migration source is FTP, OBS, or SFTP, **Source Directory/File** can contain a maximum of 50 file names, which are separated by vertical bars (|). You can also customize a file separator.

NOTE

1. CDM supports incremental file migration (by skipping repeated files), but does not support resumable transfer.
   For example, if three files are to be migrated and the second file fails to be migrated due to the network fault. When the migration task is started again, the first file is skipped. The second file, however, cannot be migrated from the point where the fault occurs, but can only be migrated again.
2. During file migration, a single task supports millions of files. If there are too many files in the directory to be migrated, you are advised to split the files into different directories and create multiple tasks.

4.9.9 Regular Expressions for Separating Semi-structured Text

During table/file migration, CDM uses delimiters to separate fields in CSV files. However, delimiters cannot be used in complex semi-structured data because the field values also contain delimiters. In this case, the regular expression can be used to separate the fields.

The regular expression is configured in **Source Job Configuration**. The migration source must be an object storage or file system, and **File Format** must be **CSV**.

Figure 4-67 Setting regular expression parameters

During the migration of CSV files, CDM can use regular expressions to separate fields and write parsed results to the migration destination. For details about the syntax of the regular expression, refer to the related documents. This section describes the regular expressions of the following log files:

- **Log4J Log**
- **Log4J Audit Log**
- **Tomcat Log**
- **Django Log**
- **Apache Server Log**

Log4J Log

- Log sample:
  ```
  2018-01-11 08:50:59,001 INFO
  [org.apache.sqoop.core.SqoopConfiguration.configureClassLoader(SqoopConfiguration.java:251)]
  Adding jars to current classloader from property: org.apache.sqoop.classpath.extra
  ```
- Regular expression:
  ```
  ^(\d.*\d) (\w*)  \[(.*)\] (\w.*).*
  ```
- Parsing result:

Table 4-107 Log4J log parsing result

Column Number	Example Value
1	2018-01-11 08:50:59,001
2	INFO
3	org.apache.sqoop.core.SqoopConfiguration.configureClassLoader(SqoopConfiguration.java:251)
4	Adding jars to current classloader from property: org.apache.sqoop.classpath.extra

Log4J Audit Log

- Log sample:
  ```
  2018-01-11 08:51:06,156 INFO
  [org.apache.sqoop.audit.FileAuditLogger.logAuditEvent(FileAuditLogger.java:61)]
  user=sqoop.anonymous.user    ip=189.xxx.xxx.75    op=show    obj=version    objId=x
  ```
- Regular expression:
  ```
  ^(\d.*\d) (\w*) \[(.*)\] user=(\w.*)    ip=(\w.*)    op=(\w.*)    obj=(\w.*)    objId=(.*).*
  ```
- Parsing result:

  Table 4-108 Log4J audit log parsing result

Column Number	Example Value
1	2018-01-11 08:51:06,156
2	INFO
3	org.apache.sqoop.audit.FileAuditLogger.logAuditEvent(FileAuditLogger.java:61)
4	sqoop.anonymous.user
5	189.xxx.xxx.75
6	show
7	version
8	x

Tomcat Log

- Log sample:
  ```
  11-Jan-2018 09:00:06.907 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log OS
  Name:               Linux
  ```
- Regular expression:
  ```
  ^(\d.*\d) (\w*) \[(.*)\] ([\w\.]*) (\w.*).*
  ```
- Parsing result:

  Table 4-109 Tomcat log parsing result

Column Number	Example Value
1	11-Jan-2018 09:00:06.907
2	INFO
3	main

Column Number	Example Value
4	org.apache.catalina.startup.VersionLoggerListener.log
5	OS Name:Linux

Django Log

- Log sample:
  ```
  [08/Jan/2018 20:59:07 ] settings     INFO     Welcome to Hue 3.9.0
  ```
- Regular expression:
  ```
  ^\[(.*)\] (\w*)     (\w*)     (.*).*
  ```
- Parsing result:

Table 4-110 Django log parsing result

Column Number	Example Value
1	08/Jan/2018 20:59:07
2	settings
3	INFO
4	Welcome to Hue 3.9.0

Apache Server Log

- Log sample:
  ```
  [Mon Jan 08 20:43:51.854334 2018] [mpm_event:notice] [pid 36465:tid 140557517657856] AH00489: Apache/2.4.12 (Unix) OpenSSL/1.0.1t configured -- resuming normal operations
  ```
- Regular expression:
  ```
  ^\[(.*)\] \[(.*)\] \[(.*)\] (.*).*
  ```
- Parsing result:

Table 4-111 Apache server log parsing result

Column Number	Example Value
1	Mon Jan 08 20:43:51.854334 2018
2	mpm_event:notice

Column Number	Example Value
3	pid 36465:tid 140557517657856
4	AH00489: Apache/2.4.12 (Unix) OpenSSL/1.0.1t configured -- resuming normal operations

4.9.10 Recording the Time When Data Is Written to the Database

When you create a job on the CDM console to migrate tables or files of a relational database, you can add a field to record the time when they were written to the database.

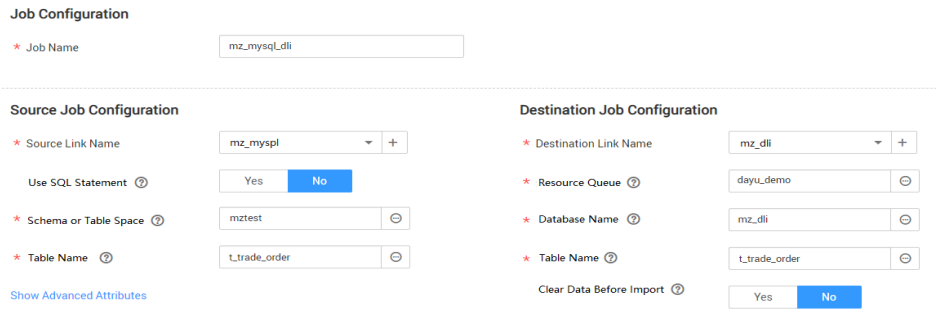
Prerequisites

- A link has been created, and the source end of the connector is a relational database.
- The destination data table contains a date and time field or timestamp field. In the automatic table creation scenario, you need to manually create the date and time field or timestamp field in the destination table in advance.

Creating a Table/File Migration Job

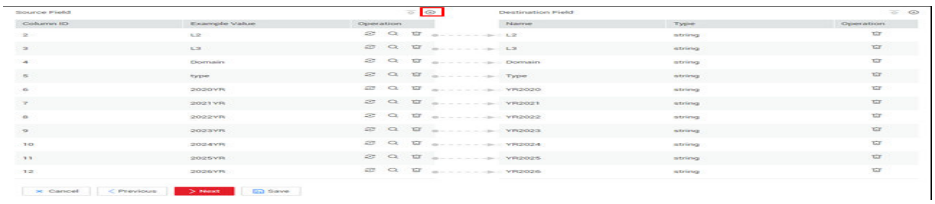
Step 1 Create a table/file migration job, and select the created source connector and destination connector.

Figure 4-68 Configuring the job



Step 2 Click **Next** to go to the **Map Field** page and click .

Figure 4-69 Configuring field mapping



Step 3 Click the **Custom Fields** tab, set the field name and value, and click **OK**.

Name: Enter **InputTime**.

Value: Enter **${timestamp()}**. For more time macro variables, see **Table 4-112**.

Figure 4-70 Add Field

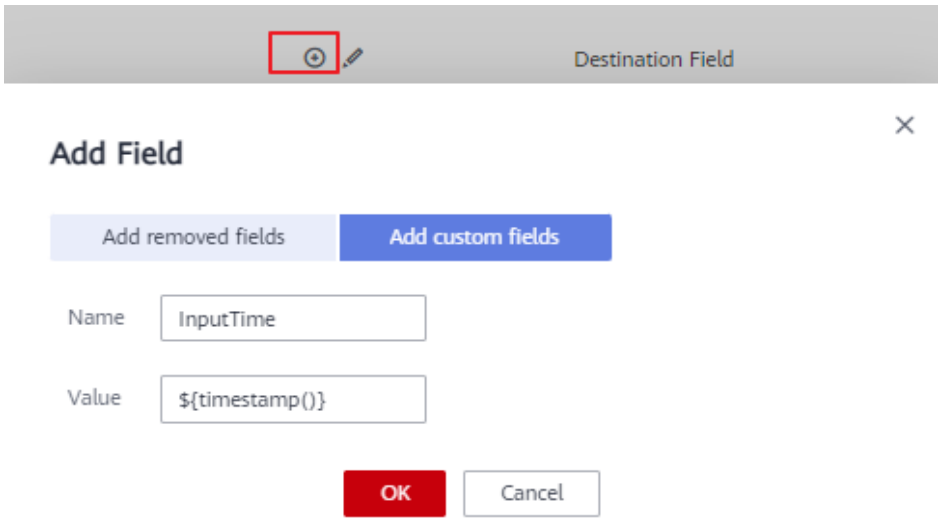


Table 4-112 Macro variable definition of time and date

Macro Variable	Description	Display Effect
${dateformat(yyyy-MM-dd)}	Returns the current date in **yyyy-MM-dd** format.	2017-10-16
${dateformat(yyyy/MM/dd)}	Returns the current date in **yyyy/MM/dd** format.	2017/10/16
${dateformat(yyyy_MM_dd HH:mm:ss)}	Returns the current time in **yyyy_MM_dd HH:mm:ss** format.	2017_10_16 09:00:00
${dateformat(yyyy-MM-dd HH:mm:ss, -1, DAY)}	Returns the current time in **yyyy-MM-dd HH:mm:ss** format. The date is one day before the current day.	2017-10-15 09:00:00
${timestamp()}	Returns the timestamp of the current time, that is, the number of milliseconds that have elapsed since 00:00:00 on January 1, 1970.	1508115600000
${timestamp(-10, MINUTE)}	Returns the timestamp generated 10 minutes before the current time point.	1508115000000

Macro Variable	Description	Display Effect
${timestamp(dateformat(yyyyMMdd))}	Returns the timestamp of 00:00:00 of the current day.	1508083200000
${timestamp(dateformat(yyyyMMdd,-1,DAY))}	Returns the timestamp of 00:00:00 of the previous day.	1507996800000
${timestamp(dateformat(yyyyMMddHH))}	Returns the timestamp of the current hour.	1508115600000

NOTE

- After a field is added, its sample value is not displayed on the console. This does not affect the field value transmission. CDM directly writes the field value to the destination end.
- The **Custom Fields** tab is available only when the source connector is JDBC, HBase, MongoDB, Elasticsearch, or Kafka, or the destination connector is HBase.
- After adding the fields, ensure that the customized import time field matches the field type of the destination table.

Step 4 Click **Next** and set task parameters. Generally, retain the default values of all parameters.

Step 5 Click **Save and Run**. On the **Table/File Migration** page, you can view the job execution progress and result.

Step 6 After the job is successfully executed, in the **Operation** column of the job, click **Historical Record** to view the job's historical execution records and read/write statistics.

On the **Historical Record** page, click **Log** to view the job logs.

Step 7 Go to the destination data source to check the time when the data is imported to the database.

----End

4.9.11 File Formats

When creating a CDM job, you need to specify **File Format** in the job parameters of the migration source and destination in some scenarios. This section describes the application scenarios, subparameters, common parameters, and usage examples of the supported file formats.

- **CSV**
- **JSON**
- **Binary**
- **Common parameters**
- **Solutions to File Format Problems**

CSV

To read or write a CSV file, set **File Format** to **CSV**. The CSV format can be used in the following scenarios:

- Import files to a database or NoSQL.
- Export data from a database or NoSQL to files.

After selecting the CSV format, you can also configure the following optional sub-parameters:

1. Line Separator

2. Field Delimiter

3. Encoding Type

4. Use Quote Character

5. Use RE to Separate Fields

6. Use First Row as Header

7. File Size

1. **Line Separator**

   Character used to separate lines in a CSV file. The value can be a single character, multiple characters, or special characters. Special characters can be entered using the URL encoded characters. The following table lists the URL encoded characters of commonly used special characters.

   Table 4-113 URL encoded characters of special characters

Special Character	URL Encoded Character
Space	%20
Tab	%09
%	%25
Enter	%0d
Newline character	%0a
Start of heading\u0001 (SOH)	%01

2. **Field Delimiter**

   Character used to separate columns in a CSV file. The value can be a single character, multiple characters, or special characters. For details, see **Table 4-113**.

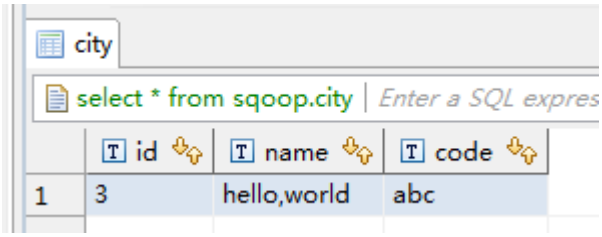
3. **Encoding Type**

   Encoding type of a CSV file. The default value is **UTF-8**. Some Chinese characters are encoded by GBK.

   If this parameter is specified at the migration source, the specified encoding type is used to parse the file. If this parameter is specified at the migration destination, the specified encoding type is used to write data to the file.

4. **Use Quote Character**
   - Exporting data from a database or NoSQL to CSV files (configuring **Use Quote Character** at the migration destination): If a field delimiter appears in the character string of a column of data at the migration source, set **Use Quote Character** to **Yes** at the migration destination to quote the character string as a whole and write it into the CSV file. Currently, CDM uses double quotation marks ("") as the quote character only. **Figure 4-71** shows that the value of the **name** field in the database contains a comma (,).

     Figure 4-71 Field value containing the field delimiter

     If you do not use the quote character, the exported CSV file is displayed as follows:

     ```
     3,hello,world,abc
     ```

     If you use the quote character, the exported CSV file is displayed as follows:

     ```
     3,"hello,world",abc
     ```

     If the data in the database contains double quotation marks ("") and you set **Use Quote Character** to **Yes**, the quote character in the exported CSV file is displayed as three double quotation marks (""""). For example, if the value of a field is **a"hello,world"c**, the exported data is as follows:

     ```
     """a"hello,world"c"""
     ```

   - Exporting CSV files to a database or NoSQL (configuring **Use Quote Character** at the migration source): If you want to import the CSV files with quoted values to a database correctly, set **Use Quote Character** to **Yes** at the migration source to write the quoted values as a whole.
5. **Use RE to Separate Fields**

   This function is used to parse complex semi-structured text, such as log files. For details, see **Using Regular Expressions to Separate Semi-structured Text**.
6. **Use First Row as Header**

   This parameter is used when CSV files are exported to other locations. If this parameter is specified at the migration source, CDM uses the first row as the header when extracting data. When the CSV files are transferred, the headers are skipped. The number of rows extracted from the migration source is more than the number of rows written to the migration destination. The log files will output the information that the header is skipped during the migration.
7. **File Size**

   This parameter is used when data is exported from the database to a CSV file. If a table contains a large amount of data, a large CSV file is generated after migration, which is inconvenient to download or view. In this case, you can

specify this parameter at the migration destination so that multiple CSV files with the specified size can be generated. The value of this parameter is an integer. The unit is MB.

JSON

The following describes information about the JSON format:

- **JSON Types Supported by CDM**
- **JSON Reference Node**
- **Copying Data from a JSON File**

1. **JSON types supported by CDM: JSON object and JSON array**
   - JSON object: A JSON file contains a single object or multiple objects separated/merged by rows.
     i. The following is a single JSON object:
     ```
     {
         "took" : 190,
         "timed_out" : false,
         "total" : 1000001,
         "max_score" : 1.0
     }
     ```
     ii. The following are JSON objects separated by rows:
     ```
     {"took" : 188, "timed_out" : false, "total" : 1000003, "max_score" : 1.0 }
     {"took" : 189, "timed_out" : false, "total" : 1000004, "max_score" : 1.0 }
     ```
     iii. The following are merged JSON objects:
     ```
     {
       "took": 190,
       "timed_out": false,
       "total": 1000001,
       "max_score": 1.0
     }
     {
       "took": 191,
       "timed_out": false,
       "total": 1000002,
       "max_score": 1.0
     }
     ```
   - JSON array: A JSON file is a JSON array consisting of multiple JSON objects.
     ```
     [{
         "took" : 190,
         "timed_out" : false,
         "total" : 1000001,
         "max_score" : 1.0
     },
     {
         "took" : 191,
         "timed_out" : false,
         "total" : 1000001,
         "max_score" : 1.0
     }]
     ```
2. **JSON Reference Node**

   Root node that records data. The data corresponding to the node is a JSON array. CDM extracts data from the array in the same mode. Use periods (.) to separate multi-layer nested JSON nodes.
3. **Copying Data from a JSON File**

   a. Example 1

Extract data from multiple objects that are separated or merged. A JSON file contains multiple JSON objects. The following gives an example:

```
{
  "took": 190,
  "timed_out": false,
  "total": 1000001,
  "max_score": 1.0
}
{
  "took": 191,
  "timed_out": false,
  "total": 1000002,
  "max_score": 1.0
}
{
  "took": 192,
  "timed_out": false,
  "total": 1000003,
  "max_score": 1.0
}
```

To extract data from the JSON object and write data to the database in the following formats, set **File Format** to **JSON** and **JSON Type** to **JSON object**, and then map fields.

Table 4-114 Example

took	timedOut	total	maxScore
190	false	1000001	1.0
191	false	1000002	1.0
192	false	1000003	1.0

b. Example 2

Extract data from the reference node. A JSON file contains a single JSON object, but the valid data is on a data node. The following gives an example:

```
{
  "took": 190,
  "timed_out": false,
  "hits": {
     "total": 1000001,
     "max_score": 1.0,
     "hits":
      [{
         "_id": "650612",
         "_source": {
            "name": "tom",
            "books": ["book1","book2","book3"]
         }
      },
      {
          "_id": "650616",
          "_source": {
             "name": "tom",
             "books": ["book1","book2","book3"]
          }
      },
    {
          "_id": "650618",
          "_source": {
             "name": "tom",
```



```
            "books": ["book1","book2","book3"]
          }
        }]
    }
}
```

To write data to the database in the following formats, set **File Format** to **JSON**, **JSON Type** to **JSON object**, and **JSON Reference Node** to **hits.hits**, and then map fields.

Table 4-115 Example

ID	SourceName	SourceBooks
650612	tom	["book1","book2","book3"]
650616	tom	["book1","book2","book3"]
650618	tom	["book1","book2","book3"]

c. Example 3

Extract data from the JSON array. A JSON file is a JSON array consisting of multiple JSON objects. The following gives an example:

```
[{
    "took" : 190,
    "timed_out" : false,
    "total" : 1000001,
    "max_score" : 1.0
 },
 {
    "took" : 191,
    "timed_out" : false,
    "total" : 1000002,
    "max_score" : 1.0
 }]
```

To write data to the database in the following formats, set **File Format** to **JSON** and **JSON Type** to **JSON array**, and then map fields.

Table 4-116 Example

took	timedOut	total	maxScore
190	false	1000001	1.0
191	false	1000002	1.0

d. Example 4

Configure a converter when parsing the JSON file. On the premise of **example 2**, to add the **hits.max_score** field to all records, that is, to write the data to the database in the following formats, perform the following operations:

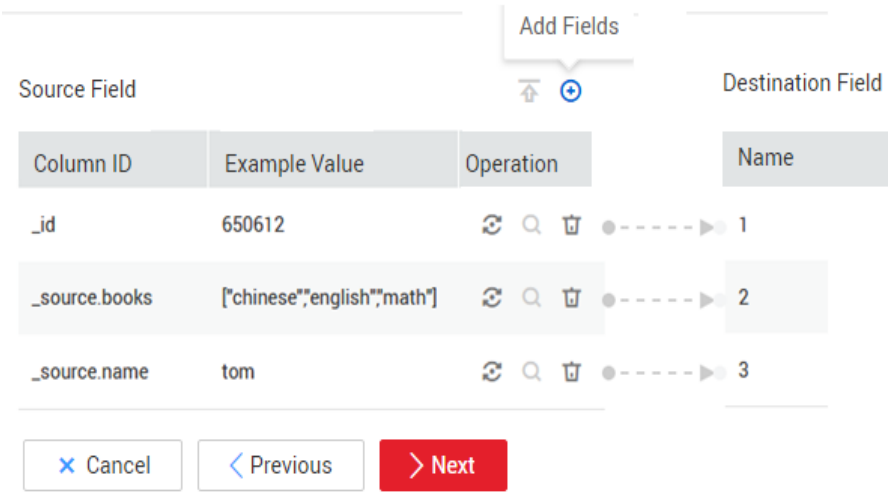
Table 4-117 Example

ID	SourceName	SourceBooks	MaxScore
650612	tom	["book1","book2","book3"]	1.0
650616	tom	["book1","book2","book3"]	1.0
650618	tom	["book1","book2","book3"]	1.0

Set **File Format** to **JSON**, **JSON Type** to **JSON object**, and **JSON Reference Node** to **hits.hits**, and then create a converter.

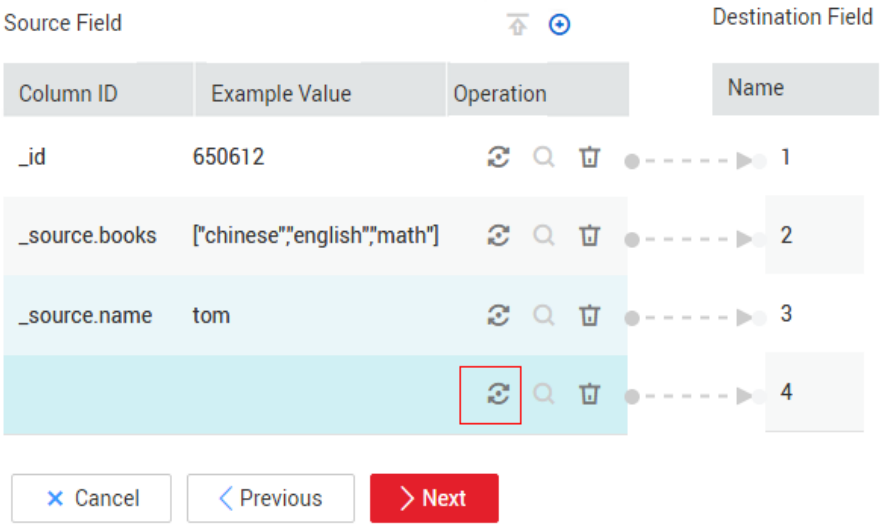
i. Click to add a field.

Figure 4-72 Adding a field



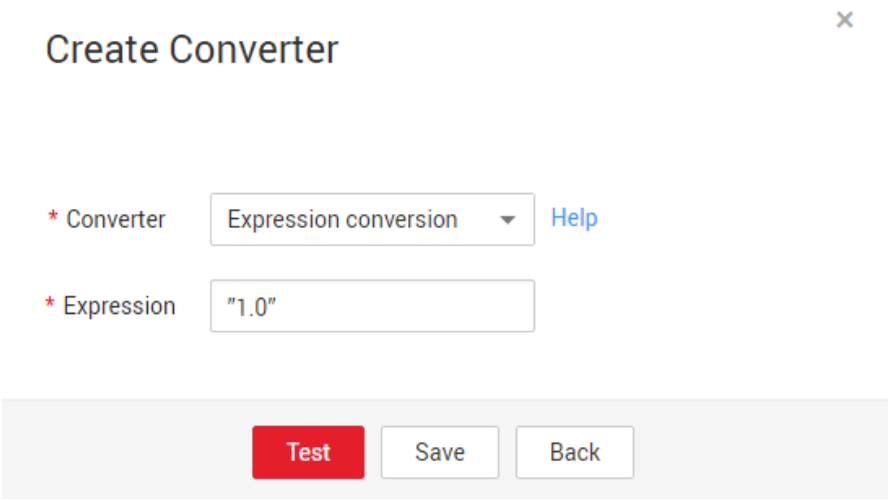
ii. Click to create a converter for the new field.

Figure 4-73 Creating a field converter



iii. Set **Converter** to **Expression conversion**, enter **"1.0"** in the **Expression** text box, and click **Save**.

Figure 4-74 Configuring a field converter



Binary

If you want to copy files between file systems, you can select the binary format. Files can be transferred in binary format at a high speed and stable performance. In addition, field mapping is not required in the second step of the job.

- **Directory structure for file transfer**

  CDM can transfer a single file or all files in a directory at a time. After the files are transferred to the migration destination, the directory structure remains unchanged.

- **Migrating incremental files**

  When you use CDM to transfer files in binary format, configure **Duplicate File Processing Method** at the migration destination for incremental file migration. For details, see **Incremental File Migration**.

  During incremental file migration, set **Duplicate File Processing Method** to **Skip**. If new files exist at the migration source or a failure occurs during the migration, run the job again, so that the migrated files will not be migrated repeatedly.

- **Write to Temporary File**

  When migrating files in binary format, you can specify whether to write the files to a temporary file at the migration destination. If this parameter is specified, the file is written to a temporary file during file replication. After the file is successfully migrated, run the **rename** or **move** command to restore the file at the migration destination.

- **Generate MD5 Hash Value**

  An MD5 hash value is generated for each transferred file, and the value is recorded in a new **.md5** file. You can specify the directory where the MD5 value is generated.

Common parameters

- **Start Job by Marker File**

  In automation scenarios, a scheduled task is configured on CDM to periodically read files from the migration source. However, files are being generated at the migration source. As a result, CDM reads data repeatedly or fails to read data from the migration source. You can specify the marker file for starting a job as **ok.txt** in the job parameters of the migration source. After the file is successfully generated at the migration source, the **ok.txt** file is generated in the file directory. In this way, CDM can read the complete file.

  In addition, you can set the suspension period. Within the suspension period, CDM periodically queries whether the marker file exists. If the file does not exist after the suspension period expires, the job fails.

  The marker file will not be migrated.

- **Job Success Marker File**

  After data is successfully migrated to a file system, an empty file is generated in the destination directory. You can specify the file name. Generally, this parameter is used together with **Start Job by Marker File**.

  The name of the job success marker file cannot be the same as that of the transferred file, for example, finish.txt. If the two files have the same name, they will overwrite each other.

- **Filter**

  When using CDM to migrate files, you can specify a filter to filter files. Files can be filtered by wildcard character or time filter.

  - If you select **Wildcard**, CDM migrates only the paths or files that meet the filter condition.
  - If you select **Time Filter**, CDM migrates only the files modified after the specified time point.

  For example, the **/table/** directory stores a large number of data table directories divided by day. **DRIVING_BEHAVIOR_20180101** to **DRIVING_BEHAVIOR_20180630** store all data of **DRIVING_BEHAVIOR** from January to June. If you only want to migrate the table data of **DRIVING_BEHAVIOR** in March, set the source directory to **/table**, filter type to wildcard, and path filter to **DRIVING_BEHAVIOR_201803***.

Solutions to File Format Problems

1. When data in a database is exported to a CSV file, if the data contains commas (,), the data in the exported CSV file is disordered.

   The following solutions are available:

   - Specify a field delimiter.

     Use a character that does not exist in the database or a rare non-printable character as the field delimiter. For example, you can set **Field Delimiter** at the destination to **%01**. In this way, the exported field delimiter is **\u0001**. For details, see **Table 4-113**.

   - Use a quote character.

     Set **Use Quote Character** to **Yes** at the migration destination. In this way, if the field in the database contains the field delimiter, CDM quotes the

field using the quote character and write the field as a whole to the CSV file.

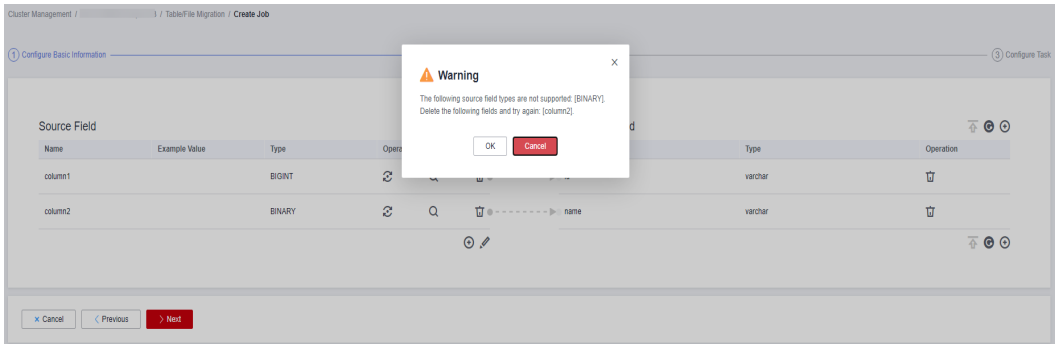
2. The data in the database contains line separators.
   - Scenario: When you use CDM to export a table in the MySQL database (a field value contains the line separator **\n**) to a CSV file, and then use CDM to import the exported CSV file to MRS HBase, data in the exported CSV file is truncated.
   - Solution: Specify a line separator.

     When you use CDM to export MySQL table data to a CSV file, set **Line Separator** at the migration destination to **%01** (ensure that the value does not appear in the field value). In this way, the line separator in the exported CSV file is **%01**. Then use CDM to import the CSV file to MRS HBase. Set **Line Separator** at the migration source to **%01**. This avoids data truncation.

4.9.12 Converting Unsupported Data Types

Scenario

When field mapping is configured on CDM, a message is displayed indicating that the data type of the field is not supported and the field needs to be deleted. If you need to use this field, you can use SQL statements to convert the field type in the source job configuration to the type supported by CDM for data migration.



Procedure

Step 1 Modify the CDM migration job and enable **Use SQL Statement**.

NOTE

The SQL statement format is as follows: **select id,cast**(*Original field name* **as INT) as** *New field name, which can be the same as the original field name* **from schemaName.tableName;**

For example, select `id`, `name`, cast(`sex` AS char(255)) AS `sex` from `test_1117869`.`test_no_support_type`;

Step 2 Wait for the fields to be converted to the data types supported by CDM.

----End

4.9.13 Auto Table Creation

Field Mapping in Automatic Table Creation

Figure 4-75 describes the field mapping between the DWS tables created by CDM and source tables. For example, if you use CDM to migrate the Oracle database to DWS, CDM automatically creates a table on DWS and maps the **NUMBER(3,0)** field of the Oracle database to the **SMALLINT** field of DWS.

Figure 4-75 Field mapping in automatic table creation

Source Database Type							Destination Database Type
Oracle	MySQL	SQL Server	PostgreSQL	Db2	GaussDB	SAP HANA	DWS
NUMBER(p,0) (p=3 or p=5)	SMALLINT,TINYINT	SMALLINT,TINYINT	SMALLINT	DECIMAL	SMALLINT	SMALLINT,TINYINT	SMALLINT
NUMBER(10,0)	INT	INT	INTEGER	INT	INTEGER	INTEGER	INTEGER
NUMBER(19,0)	BIGINT	BIGINT	BIGINT	DECIMAL	BIGINT	BIGINT	BIGINT
None	None	None	OID	None	OID	CHAR(128)	OID
NUMBER(p,s) (0 < p <= 38)	DECIMAL(p,s) (0 < p <= 65)	DECIMAL(p,s) (0 < p <= 30)	NUMERIC(p,s) (p <= 1000)	DECIMAL	NUMERIC(p,s) (p <= 1000)	DECIMAL(p,s) (0 < p <= 38)	NUMERIC(p,s) (p <= 1000)
RAW	BINARY	BINARY	BYTEA	BINARY	BYTEA	BINARY	BYTEA
CHAR	CHAR	CHAR	CHAR	CHAR	CHAR	CHAR(p) (p <= 2000)	CHAR
NCHAR	NCHAR	NCHAR	NCHAR	NCHAR	NCHAR	NCHAR(p) (p <= 5000)	NCHAR
DATE	DATE	DATE	DATE	DATE	DATE	DATE	DATE
DATE	DATETIME	DATETIME2	TIMESTAMP	TIMESTAMP	TIMESTAMP	TIMESTAMP	TIMESTAMP
VARCHAR2(p) (p <= 4000)	VARCHAR	VARCHAR(p) (if p >= 8000 p=max)	VARCHAR(p) (p <= 10485760)	VARCHAR	VARCHAR(p) (p <= 10485760)	VARCHAR(p) (p <= 5000)	VARCHAR(p) (p <= 10485760)
FLOAT	DOUBLE	FLOAT	DOUBLE PRECISION	FLOAT	DOUBLE PRECISION	DOUBLE	DOUBLE PRECISION
FLOAT	REAL	FLOAT	REAL	FLOAT	REAL	REAL	REAL
CLOB	TEXT	TEXT	TEXT	TEXT	TEXT	CLOB	TEXT
DATE	None	TIME	TIME	TIME	TIME	TIME	TIME
BOOLEAN	None	None	BOOLEAN	BOOLEAN	BOOLEAN	BOOLEAN	BOOLEAN

Table 4-118, **Table 4-119**, **Table 4-120**, and **Table 4-121** describe the field type mapping between Hive tables and source tables when CDM automatically creates tables in Hive. For example, if you use CDM to migrate the MySQL database to Hive, CDM automatically creates a table on Hive and maps the **YEAR** field of the MySQL database to the **DATE** field of Hive.

NOTE

- For the DECIMAL type, if the length of the source data exceeds the Hive length, the precision may be lost.
- For the DECIMAL type, the precision is greater than or equal to 1 and less than or equal to 38, and the scale is greater than or equal to 0. If the precision for the source is greater than 38 bits, the precision for Hive table creation is 38 bits. If the scale is less than 0, the scale for Hive table creation is 0. In this case, precision loss may occur after data is written.

Table 4-118 Field mapping in automatic table creation for MySQL-to-Hive migration

Data Type (MySQL)	Data Type (Hive)	Description
Value		
tinyint(1), bit(1)	BOOLEAN	-

Data Type (MySQL)	Data Type (Hive)	Description
TINYINT	SMALLINT	-
TINYINT UNSIGNED	SMALLINT	-
SMALLINT	SMALLINT	-
SMALLINT UNSIGNED	INTEGER	-
MEDIUMINT	INTEGER	-
MEDIUMINT UNSIGNED	BIGINT	-
INT	INTEGER	-
INT UNSIGNED	BIGINT	-
BIGINT	BIGINT	-
BIGINT UNSIGNED	DECIMAL(38,0)	-
DECIMAL(P,S)	DECIMAL(P,S)	The MySQL database supports a maximum of 65 bits. For Hive, the precision is greater than or equal to 1 and less than or equal to 38, and the scale is greater than or equal to 0. If the precision for the MySQL database is greater than 38 bits, the precision for Hive table creation is 38 bits. If the scale is less than 0, the scale for Hive table creation is 0.
FLOAT	FLOAT	-
FLOAT UNSIGNED	FLOAT	-
DOUBLE	DOUBLE	-
DOUBLE UNSIGNED	DOUBLE	-
Time		
DATE	DATE	-
YEAR	DATE	-
DATETIME	TIMESTAMP	-
TIMESTAMP	TIMESTAMP	-
TIME	STRING	-

Data Type (MySQL)	Data Type (Hive)	Description
Character		
CHAR(N)	CHAR(N*3)	If the value of (n*3<255) is greater than 255 (CHAR_MAX_LENGTH), varchar(N*3) is created. If the value of (n*3<255) is greater than 65535 (VARCHAR_MAX_LENGTH), a string is created.
VARCHAR(N)	VARCHAR(N*3)	If the value is greater than 65535 (VARCHAR_MAX_LENGTH), a string is created.
BINARY	BINARY	-
VARBINARY	BINARY	-
TINYBLOB	BINARY	-
MEDIUMBLOB	BINARY	-
BLOB	BINARY	-
LONGBLOB	BINARY	-
TINYTEXT	VARCHAR(765)	-
MEDIUMTEXT	STRING	-
TEXT	STRING	-
LONGTEXT	STRING	-
Others	STRING	-

Table 4-119 Field mapping in automatic table creation for Oracle-to-Hive migration

Data Type (Oracle)	Data Type (Hive)	Description
Character		
CHAR(N)	CHAR(N*3)	If the value of (n*3<255) is greater than 255 (CHAR_MAX_LENGTH), varchar(N*3) is created. If the value of (n*3<255) is greater than 65535 (VARCHAR_MAX_LENGTH), a string is created.

Data Type (Oracle)	Data Type (Hive)	Description
VARCHAR(N)	VARCHAR(N*3)	If the value is greater than 65535 (VARCHAR_MAX_LENGTH), a string is created.
VARCHAR2	VARCHAR(N*3)	If the value is greater than 65535 (VARCHAR_MAX_LENGTH), a string is created.
NCHAR	CHAR(N*3)	-
NVARCHAR2	STRING	-
Value		
NUMBER	DECIMAL(P,S)	For Hive, the precision is greater than or equal to 1 and less than or equal to 38, and the scale is greater than or equal to 0. If the precision for the MySQL database is greater than 38 bits, the precision for Hive table creation is 38 bits. If the scale is less than 0, the scale for Hive table creation is 0.
BINARY_FLOAT	FLOAT	-
BINARY_DOUBLE	DOUBLE	-
FLOAT	FLOAT	-
Time		
DATE	TIMESTAMP	-
TIMESTAMP	TIMESTAMP	-
TIMESTAMP WITH TIME ZONE	STRING	-
TIMESTAMP WITH LOCAL TIME ZONE	STRING	-
INTERVAL	STRING	-
Binary		
BLOB	BINARY	-
CLOB	STRING	-
NCLOB	STRING	-
LONG	STRING	-
LONG_RAW	BINARY	-

Data Type (Oracle)	Data Type (Hive)	Description
RAW	BINARY	-
Other	STRING	-

Table 4-120 Field mapping in automatic table creation for PostgreSQL/DWS-to-Hive migration

Data Type (PostgreSQL/ DWS)	Data Type (Hive)	Description
Value		
int2	SMALLINT	-
int4	INT	-
int8	BIGINT	-
real	FLOAT	-
float4	FLOAT	-
float8	DOUBLE	-
smallserial	SMALLINT	-
serial	INT	-
bigserial	BIGINT	-
numeric(p,s)	DECIMAL(P,S)	For Hive, the precision is greater than or equal to 1 and less than or equal to 38, and the scale is greater than or equal to 0. If the precision for the MySQL database is greater than 38 bits, the precision for Hive table creation is 38 bits. If the scale is less than 0, the scale for Hive table creation is 0.
money	DOUBLE	-
bit(1)	TINYINT	-
varbit	STRING	-
Character		
varchar(n)	VARCHAR(N*3)	If the value is greater than 65535 (VARCHAR_MAX_LENGTH), a string is created.

Data Type (PostgreSQL/ DWS)	Data Type (Hive)	Description
bpchar(n)	CHAR(N*3)	If the value of (n*3<255) is greater than 255 (CHAR_MAX_LENGTH), varchar(N*3) is created. If the value of (n*3<255) is greater than 65535 (VARCHAR_MAX_LENGTH), a string is created.
char(n)	CHAR(N*3)	If the value of (n*3<255) is greater than 255 (CHAR_MAX_LENGTH), varchar(N*3) is created. If the value of (n*3<255) is greater than 65535 (VARCHAR_MAX_LENGTH), a string is created.
bytea	BINARY	-
text	STRING	-
Time		
interval	STRING	-
date	DATE	-
time	STRING	-
timetz	STRING	-
timestamp	TIMESTAMP	-
timestamptz	TIMESTAMP	-
Boolean		
bool	BOOLEAN	-
Other	STRING	-

Table 4-121 Field mapping in automatic table creation for SQL Server-to-Hive migration

Data Type (SQL Server)	Data Type (Hive)	Description
Value		
TINYINT	SMALLINT	-
SMALLINT	SMALLINT	-
INT	INT	-

Data Type (SQL Server)	Data Type (Hive)	Description
BIGINT	BIGINT	-
DECIMAL	DECIMAL(P,S)	For Hive, the precision is greater than or equal to 1 and less than or equal to 38, and the scale is greater than or equal to 0. If the precision for the MySQL database is greater than 38 bits, the precision for Hive table creation is 38 bits. If the scale is less than 0, the scale for Hive table creation is 0.
NUMERIC	DECIMAL(P,S)	For Hive, the precision is greater than or equal to 1 and less than or equal to 38, and the scale is greater than or equal to 0. If the precision for the MySQL database is greater than 38 bits, the precision for Hive table creation is 38 bits. If the scale is less than 0, the scale for Hive table creation is 0.
FLOAT	DOUBLE	-
REAL	FLOAT	-
SMALLMONEY	DECIMAL(10,4)	-
MONEY	DECIMAL(19,4)	-
BIT(1)	TINYINT	-
Time		
DATE	DATE	-
DATETIME	TIMESTAMP	-
DATETIME2	TIMESTAMP	-
DATETIMEOFFSET	STRING	-
TIME(p)	STRING	-
TIMESTAMP	BINARY	-
Character		
CHAR(n)	CHAR(n*3)	If the value of (n*3<255) is greater than 255 (CHAR_MAX_LENGTH), varchar(N*3) is created. If the value of (n*3<255) is greater than 65535 (VARCHAR_MAX_LENGTH), a string is created.

Data Type (SQL Server)	Data Type (Hive)	Description
VARCHAR(n)	VARCHAR(n*3)	If the value of (n*3<255) is greater than 255 (CHAR_MAX_LENGTH), varchar(N*3) is created. If the value of (n*3<255) is greater than 65536 (VARCHAR_MAX_LENGTH), a string is created.
NCHAR(n)	VARCHAR(n*3)	If the value of (n*3<255) is greater than 255 (CHAR_MAX_LENGTH), varchar(N*3) is created. If the value of (n*3<255) is greater than 65537 (VARCHAR_MAX_LENGTH), a string is created.
NVARCHAR(n)	VARCHAR(n*3)	If the value of (n*3<255) is greater than 255 (CHAR_MAX_LENGTH), varchar(N*3) is created. If the value of (n*3<255) is greater than 65538 (VARCHAR_MAX_LENGTH), a string is created.
Binary		
BINARY	BINARY	-
VARBINARY	BINARY	-
TEXT	STRING	-
Other	STRING	-

4.10 Tutorials

4.10.1 Creating an MRS Hive Link

MRS Hive links are applicable to the MapReduce Service (MRS). This tutorial describes how to create an MRS Hive link.

Prerequisites

- You have created a CDM cluster.
- You have obtained the Manager IP address, and administrator account and password of the MRS cluster, and the account has the permissions to import and export data.
- The MRS cluster and the CDM cluster can communicate with each other. The following requirements must be met for network interconnection:
  - If the CDM cluster and the cloud service are in different regions, a public network or a dedicated connection is required for enabling

communication between the CDM cluster and the cloud service. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, the host where the data source is located can access the Internet, and the port has been enabled in the firewall rules.

- If the CDM cluster and the cloud service are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**.
- The cloud service instance and the CDM cluster belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.

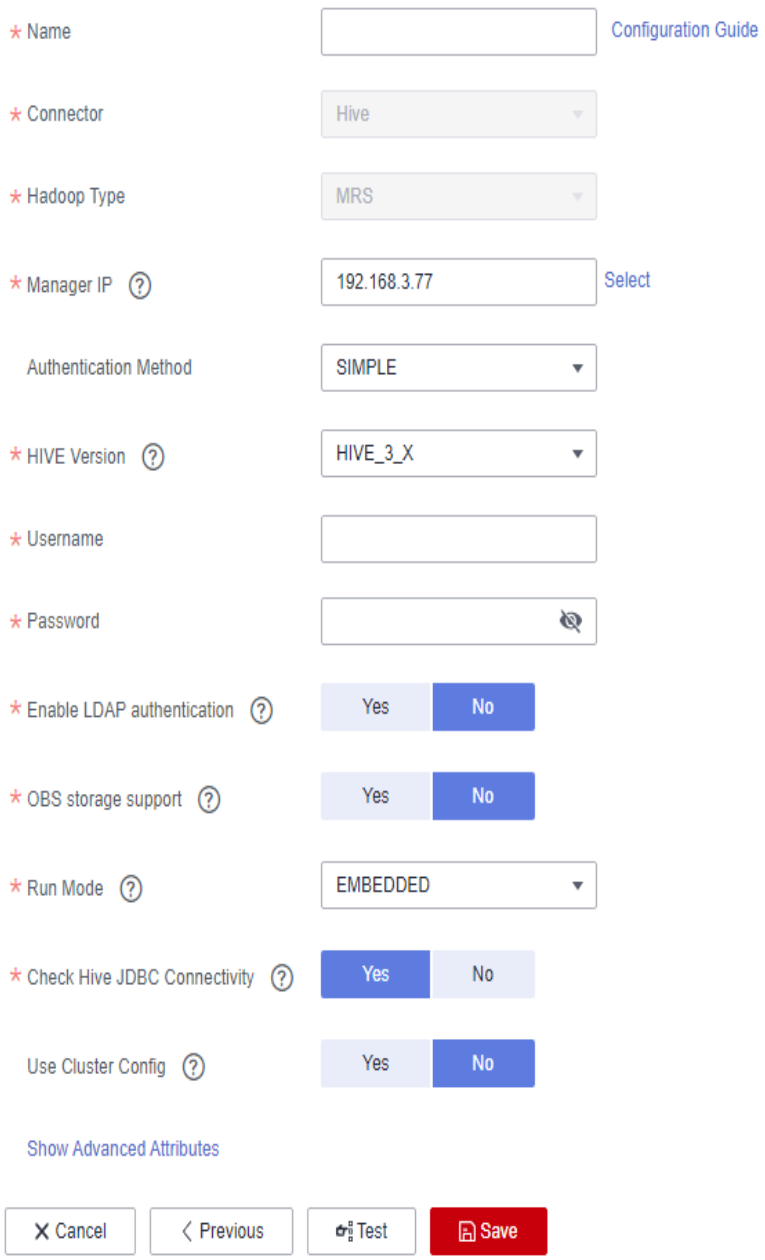
Creating an MRS Hive Link

Step 1 On the **Cluster Management** page, locate a cluster and click **Job Management** in the **Operation** column. On the displayed page, click the **Links** tab and then **Create Link**.

Figure 4-76 Selecting a connector type

Step 2 Select **MRS Hive** and click **Next** to configure parameters for the MRS Hive link.

Figure 4-77 Creating an MRS Hive link



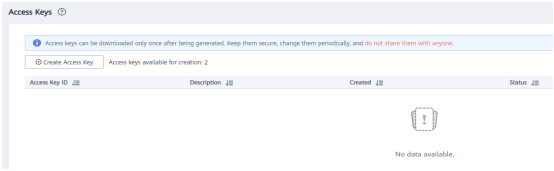
Step 3 Click **Show Advanced Attributes** to view more optional parameters. Retain their default values. The following table lists the mandatory parameters.

Table 4-122 MRS Hive link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	hivelink

Parameter	Description	Example Value
Manager IP	Floating IP address of MRS Manager. Click **Select** next to the **Manager IP** text box to select an MRS cluster. CDM automatically fills in the authentication information.	127.0.0.1
Authentication Method	Authentication method used for accessing MRS ● **SIMPLE**: Select this for non-security mode. ● **KERBEROS**: Select this for security mode.	SIMPLE
HIVE Version	Set this to the Hive version on the server.	HIVE_3_X
Username	If **Authentication Method** is set to **KERBEROS**, you must provide the username and password used for logging in to MRS Manager. If you need to create a snapshot when exporting a directory from HDFS, the user configured here must have the administrator permission on HDFS. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user and set **Username** and **Password** to the username and password of the created MRS user when creating an MRS data connection. **NOTE** ● If the CDM cluster version is 2.9.0 or later and the MRS cluster version is 3.1.0 or later, the created user must have the permissions of the **Manager_viewer** role to create links on CDM. To perform operations on databases, tables, and columns of an MRS component, you also need to add the database, table, and column permissions of the MRS component to the user by following the instructions in the MRS documentation. ● If the CDM cluster version is earlier than 2.9.0 or the MRS cluster version is earlier than 3.1.0, the created user must have the permissions of **Manager_administrator** or **System_administrator** to create links on CDM. ● A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.	cdm
Password	Password used for logging in to MRS Manager	-

Parameter	Description	Example Value
Enable ldap	This parameter is available when **Proxy connection** is selected for **Connection Type**. If LDAP authentication is enabled for an external LDAP server connected to MRS Hive, the LDAP username and password are required for authenticating the connection to MRS Hive. In this case, this option must be enabled. Otherwise, the connection will fail.	No
ldapUsername	This parameter is mandatory when **Enable ldap** is enabled. Enter the username configured when LDAP authentication was enabled for MRS Hive.	-
ldapPassword	This parameter is mandatory when **Enable ldap** is enabled. Enter the password configured when LDAP authentication was enabled for MRS Hive.	-
OBS storage support	The server must support OBS storage. When creating a Hive table, you can store the table in OBS.	No

Parameter	Description	Example Value
AK	This parameter is mandatory when **OBS storage support** is enabled. The account corresponding to the AK/SK pair must have the OBS Buckets Viewer permission. Otherwise, OBS cannot be accessed and the "403 AccessDenied" error is reported. You need to create an access key for the current account and obtain an AK/SK pair. 1. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list. 2. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 4-78**. **Figure 4-78** Clicking Create Access Key  3. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**. **NOTE** – Only two access keys can be added for each user. – To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.	-
SK		-

Parameter	Description	Example Value
Run Mode	This parameter is used only when the Hive version is **HIVE_3_X**. Possible values are: • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **Standalone**: The link instance runs in an independent process. If CDM needs to connect to multiple Hadoop data sources (MRS, Hadoop, or CloudTable) with both Kerberos and Simple authentication modes, **Standalone** prevails. **NOTE** The **STANDALONE** mode is used to solve the version conflict problem. If the connector versions of the source and destination ends of the same link are different, a JAR file conflict occurs. In this case, you need to place the source or destination end in the STANDALONE process to prevent the migration failure caused by the conflict.	EMBEDDED
Check Hive JDBC Connectivity	Whether to check the Hive JDBC connectivity	No
Use Cluster Config	You can use the cluster configuration to simplify parameter settings for the Hadoop connection.	No
Cluster Config Name	This parameter is valid only when **Use Cluster Config** is set to **Yes**. Select a cluster configuration that has been created. For details about how to configure a cluster, see **Managing Cluster Configurations**.	hive_01

NOTE

Click **Show Advanced Attributes**, and then click **Add** to add configuration attributes of other clients. The name and value of each attribute must be configured. You can click **Delete** to delete no longer used attributes.

Step 4 Click **Save** to return to the **Links**page.

----End

4.10.2 Creating a MySQL Link

MySQL links are applicable to third-party cloud MySQL services and MySQL created in a local data center or ECS. This tutorial describes how to create a MySQL link.

Prerequisites

- You have obtained the IP address, port, database name, username, and password for connecting to the MySQL database. In addition, the user must have the read and write permissions on the MySQL database.
- The on-premises MySQL database can be accessed through the public network. If the MySQL database is deployed on an on-premises data center or a third-party cloud, ensure that an IP address that can be accessed from the public network has been configured for the MySQL database, or the VPN or Direct Connect between the on-premises data center and the cloud service platform has been established.
- You have created a CDM cluster.

Creating a MySQL Link

Step 1 Access the CDM console, choose **Cluster Management** in the navigation pane, locate the target cluster, and choose **Job Management** > **Link Management** > **Driver Management**. The **Driver Management** page is displayed.

Step 2 On the **Driver Management** page, click the document link in the **Recommended Version** column of the MySQL driver and obtain the driver file as instructed.

Step 3 On the **Driver Management** page, upload the MySQL driver using either of the following methods:

Click **Upload** in the **Operation** column and select a local driver.

Alternatively, click **Copy from SFTP** in the **Operation** column and configure the **SFTP Link** name and **Driver File Path**.

Step 4 On the **Cluster Management** page, click **Job Management** of the cluster and choose **Links** > **Create Link** to enter the page for selecting the connector.

Figure 4-79 Selecting a connector type

Step 5 Select **MySQL** and click **Next** to configure parameters for the MySQL link.

Table 4-123 MySQL link parameters

Parameter	Description	Example Value
Name	Enter a unique link name.	mysqllink
Database Server	IP address or domain name of the MySQL database	192.168.1.110
Port	MySQL database port	3306
Database Name	Name of the MySQL database	sqoop
Username	User who has the read, write, and delete permissions on the MySQL database	admin
Password	Password of the user	-
Use Local API	Whether to use the local API of the database for acceleration. (The system attempts to enable the **local_infile** system variable of the MySQL database.)	Yes
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-
local_infile Character Set	When using local_infile to import data to MySQL, you can configure the encoding format.	utf8
Driver Version	A driver version that adapts to MySQL	-
Agent	The agent function will be unavailable soon and does not need to be configured.	-
Fetch Size	Number of rows obtained by each request	1000
Commit Size	(Optional) Displayed when you click **Show Advanced Attributes**. Number of records submitted each time. Set this parameter based on the destination and data size of the job. If the value is too large or too small, the job execution time may be affected.	1000
Link Attributes	Custom attributes of the link	useCompression=true

Parameter	Description	Example Value
Reference Sign	Delimiter used to separate referenced table names or column names This parameter is left blank by default.	'
Batch Size	Number of rows written each time. It should be less than **Commit Size**. When the number of rows written reaches the value of **Commit Size**, the rows will be committed to the database.	100

Step 6 Click **Save** to return to the **Links** page.

NOTE

If an error occurs during the saving, the security settings of the MySQL database are incorrect. In this case, you need to enable the EIP of the CDM cluster to access the MySQL database.

----End

4.10.3 Migrating Data from MySQL to MRS Hive

MRS provides enterprise-level big data clusters on the cloud. It contains HDFS, Hive, and Spark components and is applicable to massive data analysis of enterprises.

Hive supports SQL to help users perform extraction, transformation, and loading (ETL) operations on large-scale data sets. Query on large-scale data sets takes a long time. In many scenarios, you can create Hive partitions to reduce the total amount of data to be scanned each time. This significantly improves query performance.

Hive partitions are implemented by using the HDFS subdirectory function. Each subdirectory contains the column names and values of each partition. If there are multiple partitions, many HDFS subdirectories exist. It is not easy to load external data to each partition of the Hive table without relying on tools. With CDM, you can easily load data of the external data sources (relational databases, object storage services, and file system services) to Hive partition tables.

This section describes how to migrate data from the MySQL database to the MRS Hive partition table.

Scenario

Suppose that there is a **trip_data** table in the MySQL database. The table stores cycling records such as the start time, end time, start sites, end sites, and rider IDs. For details about the fields in the **trip_data** table, see **Figure 4-80**.

Figure 4-80 MySQL table fields

Column Name	#	Data Type
TripID	1	int(11)
Duration	2	int(11)
StartDate	3	timestamp
StartStation	4	varchar(64)
StartTerminal	5	int(11)
EndDate	6	timestamp
EndStation	7	varchar(64)
EndTerminal	8	int(11)
Bike	9	int(11)
SubscriberType	10	varchar(32)
ZipCodev	11	varchar(10)

The following describes how to use CDM to import the **trip_data** table in the MySQL database to the MRS Hive partition table. The procedure is as follows:

1. **Creating a Hive Partition Table on MRS Hive**
2. **Creating a CDM Cluster and Binding an EIP to the Cluster**
3. **Creating a MySQL Link**
4. **Creating a Hive Link**
5. **Creating a Migration Job**

Prerequisites

- MRS is available.
- You have obtained the IP address, port, database name, username, and password for connecting to the MySQL database. In addition, the user must have the read and write permissions on the MySQL database.
- You have uploaded the MySQL database driver on the **Job Management** > **Links** > **Driver Management** page.

Creating a Hive Partition Table on MRS Hive

On MRS Hive, run the following SQL statement to create a Hive partition table named **trip_data** with three new fields **y**, **ym**, and **ymd** used as partition fields. The SQL statement is as follows:

```
create table trip_data(TripID int,Duration int,StartDate timestamp,StartStation varchar(64),StartTerminal int,EndDate timestamp,EndStation varchar(64),EndTerminal int,Bike int,SubscriberType varchar(32),ZipCodev varchar(10))partitioned by (y int,ym int,ymd int);
```

NOTE

The **trip_data** partition table has three partition fields: year, year and month, and year, month, and date of the start time of a ride. For example, if the start time of a ride is **2018/5/11 9:40**, the record is saved in the **trip_data/2018/201805/20180511** partition. When the records in the **trip_data** table are summarized, only part of the data needs to be scanned, greatly improving the performance.

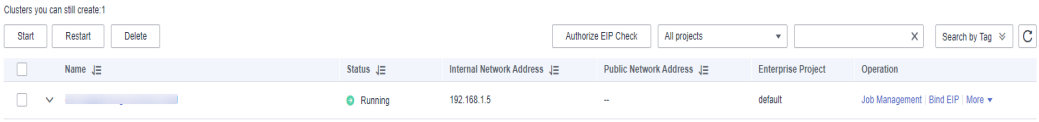
Creating a CDM Cluster and Binding an EIP to the Cluster

Step 1 The key configurations are as follows:

- The flavor of the CDM cluster is selected based on the amount of data to be migrated. Generally, cdm.medium meets the requirements for most migration scenarios.
- The CDM and MRS clusters must be in the same VPC, subnet, and security group.

Step 2 After the CDM cluster is created, on the **Cluster Management** page, click **Bind EIP** in the **Operation** column to bind an EIP to the cluster. The CDM cluster uses the EIP to access MySQL.

Figure 4-81 Cluster list



NOTE

If SSL encryption is configured for the access channel of a local data source, CDM cannot connect to the data source using the EIP.

----End

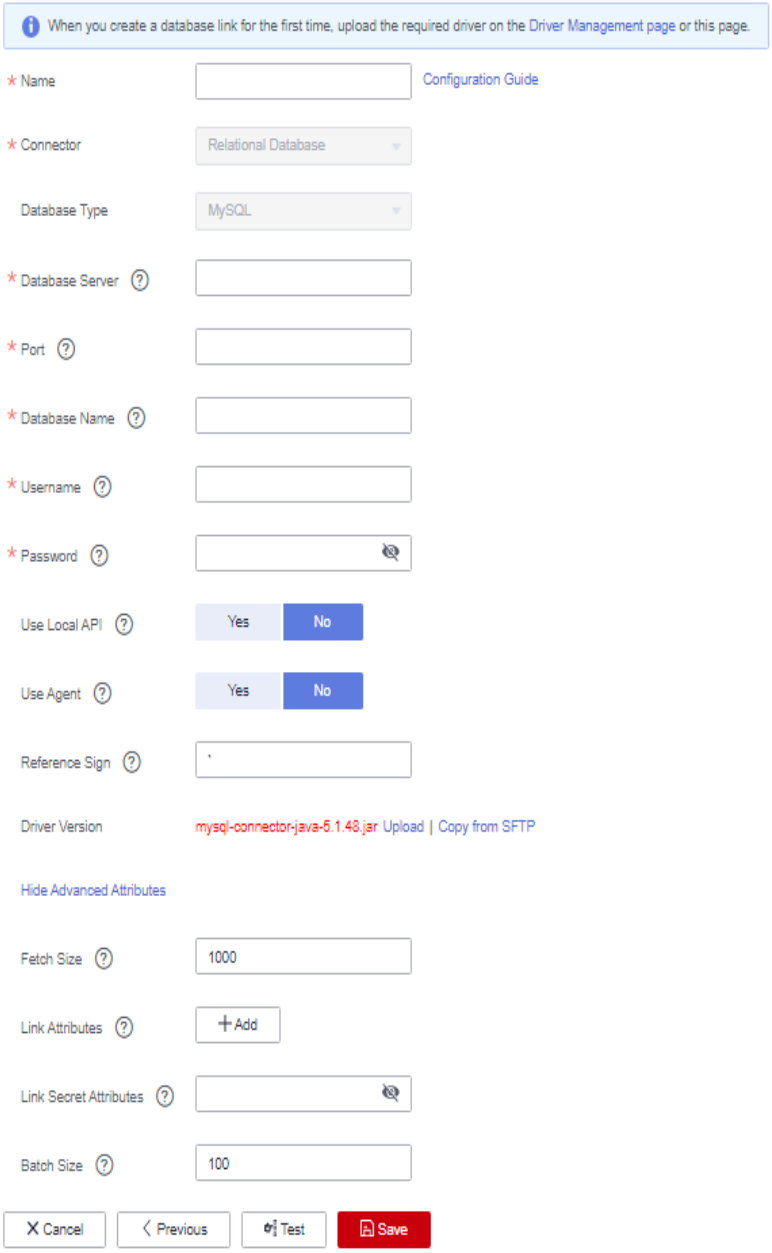
Creating a MySQL Link

Step 1 On the **Cluster Management** page, locate a cluster and click **Job Management** in the **Operation** column. On the displayed page, click the **Links** tab and then **Create Link**.

Figure 4-82 Selecting a connector

Step 2 Select **RDS for MySQL** and click **Next** to set the link parameters.

Figure 4-83 Creating a MySQL Link



Click **Show Advanced Attributes** to view more optional parameters. For details, see **Link to an RDS for MySQL/MySQL Database**. Retain the default values for the optional parameters and configure the mandatory parameters described in **Table 4-124**.

Table 4-124 MySQL link parameters

Parameter	Description	Example Value
Name	Enter a unique link name.	mysqllink

Parameter	Description	Example Value
Database Server	IP address or domain name of the MySQL database	N/A
Port	MySQL database port	3306
Database Name	Name of the MySQL database	sqoop
Username	User who has the read, write, and delete permissions on the MySQL database	admin
Password	Password of the user	N/A
Use Local API	Whether to use the local API of the database for acceleration. (The system attempts to enable the **local_infile** system variable of the MySQL database.)	Yes
Use Agent	Whether to extract data from the data source through an agent	No
local_infile Character Set	When using local_infile to import data to MySQL, you can configure the encoding format.	utf8
Driver Version	Before connecting CDM to a relational database, you need to upload the JDK 8 .jar driver of the relational database. Download the MySQL driver 5.1.48 from **https://downloads.mysql.com/archives/c-j/**, obtain **mysql-connector-java-5.1.48.jar**, and upload it.	N/A

Step 3 Click **Save**. The **Link Management** page is displayed.

NOTE

If an error occurs during the saving, the security settings of the MySQL database are incorrect. In this case, you need to enable the EIP of the CDM cluster to access the MySQL database.

----End

Creating a Hive Link

Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 4-84 Selecting a connector type

Step 2 Select **MRS Hive** and click **Next** to configure parameters for the MRS Hive link.

Figure 4-85 Creating an MRS Hive link

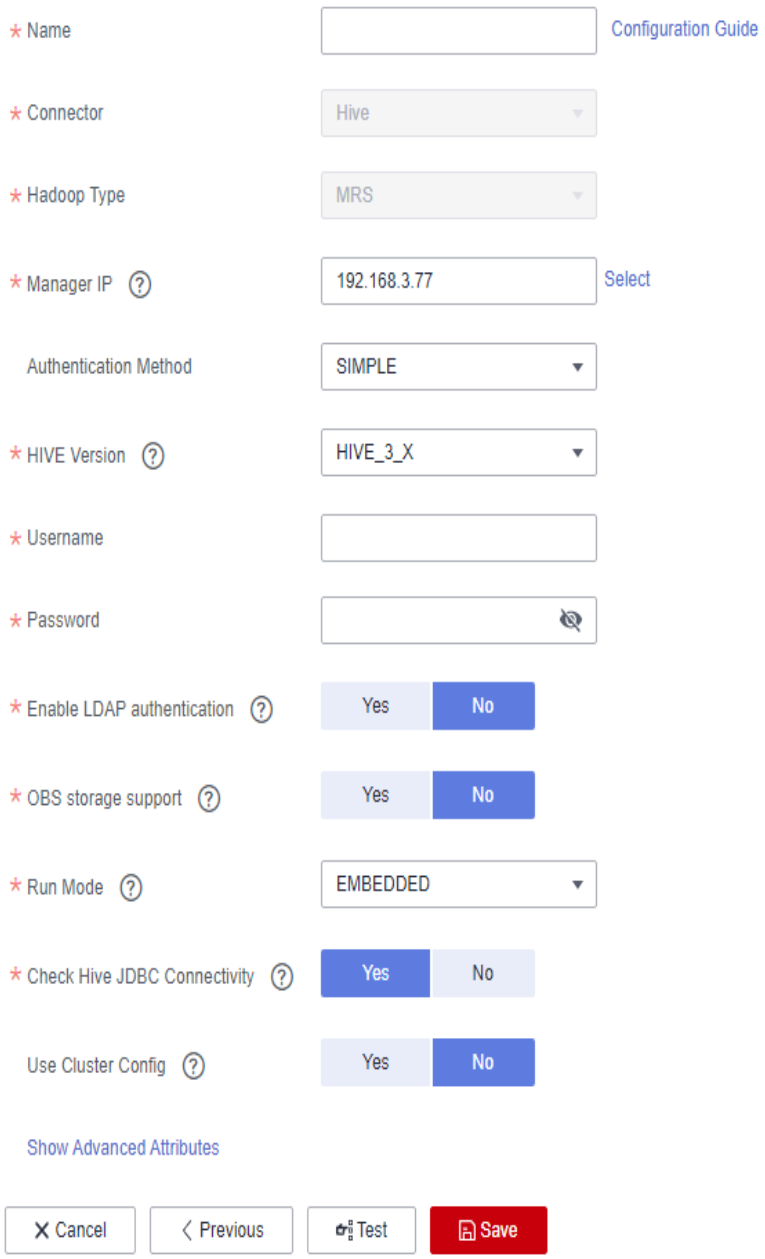


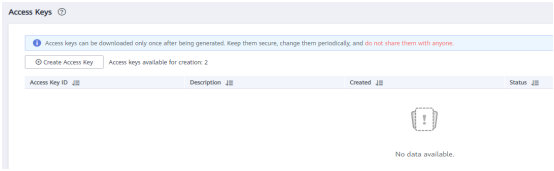
Table 4-125 lists the parameters. Configure these parameters based on your actual situation.

Table 4-125 MRS Hive link parameters

Parameter	Description	Example Value
Name	Link name, which should be defined based on the data source type, so it is easier to remember what the link is for	hivelink

Parameter	Description	Example Value
Manager IP	Floating IP address of MRS Manager. Click **Select** next to the **Manager IP** text box to select an MRS cluster. CDM automatically fills in the authentication information.	127.0.0.1
Authentication Method	Authentication method used for accessing MRS ● **SIMPLE**: Select this for non-security mode. ● **KERBEROS**: Select this for security mode.	SIMPLE
HIVE Version	Set this to the Hive version on the server.	HIVE_3_X
Username	If **Authentication Method** is set to **KERBEROS**, you must provide the username and password used for logging in to MRS Manager. If you need to create a snapshot when exporting a directory from HDFS, the user configured here must have the administrator permission on HDFS. To create a data connection for an MRS security cluster, do not use user **admin**. The **admin** user is the default management page user and cannot be used as the authentication user of the security cluster. You can create an MRS user and set **Username** and **Password** to the username and password of the created MRS user when creating an MRS data connection. **NOTE** ● If the CDM cluster version is 2.9.0 or later and the MRS cluster version is 3.1.0 or later, the created user must have the permissions of the **Manager_viewer** role to create links on CDM. To perform operations on databases, tables, and columns of an MRS component, you also need to add the database, table, and column permissions of the MRS component to the user by following the instructions in the MRS documentation. ● If the CDM cluster version is earlier than 2.9.0 or the MRS cluster version is earlier than 3.1.0, the created user must have the permissions of **Manager_administrator** or **System_administrator** to create links on CDM. ● A user with only the **Manager_tenant** or **Manager_auditor** permission cannot create connections.	cdm
Password	Password used for logging in to MRS Manager	-

Parameter	Description	Example Value
Enable ldap	This parameter is available when **Proxy connection** is selected for **Connection Type**. If LDAP authentication is enabled for an external LDAP server connected to MRS Hive, the LDAP username and password are required for authenticating the connection to MRS Hive. In this case, this option must be enabled. Otherwise, the connection will fail.	No
ldapUsername	This parameter is mandatory when **Enable ldap** is enabled. Enter the username configured when LDAP authentication was enabled for MRS Hive.	-
ldapPassword	This parameter is mandatory when **Enable ldap** is enabled. Enter the password configured when LDAP authentication was enabled for MRS Hive.	-
OBS storage support	The server must support OBS storage. When creating a Hive table, you can store the table in OBS.	No

Parameter	Description	Example Value
AK	This parameter is mandatory when **OBS storage support** is enabled. The account corresponding to the AK/SK pair must have the OBS Buckets Viewer permission. Otherwise, OBS cannot be accessed and the "403 AccessDenied" error is reported. You need to create an access key for the current account and obtain an AK/SK pair. 1. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list. 2. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 4-86**. **Figure 4-86** Clicking Create Access Key  3. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**. **NOTE** – Only two access keys can be added for each user. – To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.	-
SK		-

Parameter	Description	Example Value
Run Mode	This parameter is used only when the Hive version is **HIVE_3_X**. Possible values are: • **EMBEDDED**: The link instance runs with CDM. This mode delivers better performance. • **Standalone**: The link instance runs in an independent process. If CDM needs to connect to multiple Hadoop data sources (MRS, Hadoop, or CloudTable) with both Kerberos and Simple authentication modes, **Standalone** prevails. **NOTE** The **STANDALONE** mode is used to solve the version conflict problem. If the connector versions of the source and destination ends of the same link are different, a JAR file conflict occurs. In this case, you need to place the source or destination end in the STANDALONE process to prevent the migration failure caused by the conflict.	EMBEDDED
Check Hive JDBC Connectivity	Whether to check the Hive JDBC connectivity	No
Use Cluster Config	You can use the cluster configuration to simplify parameter settings for the Hadoop connection.	No
Cluster Config Name	This parameter is valid only when **Use Cluster Config** is set to **Yes**. Select a cluster configuration that has been created. For details about how to configure a cluster, see **Managing Cluster Configurations**.	hive_01

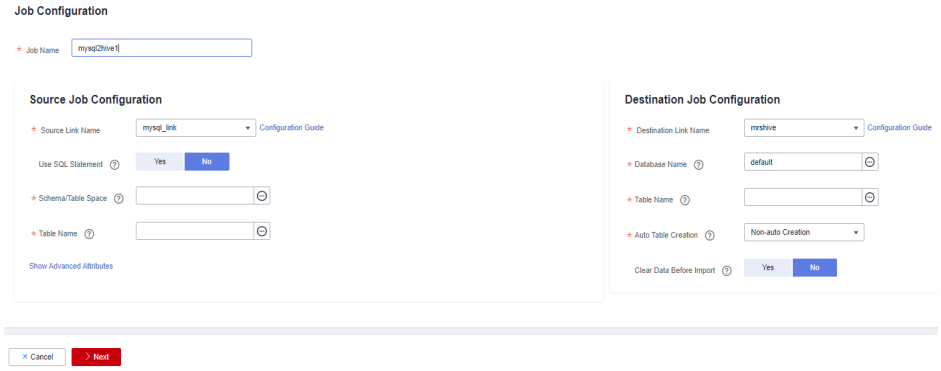
Step 3 Click **Save**. The **Link Management** page is displayed.

----End

Creating a Migration Job

Step 1 Click the **Table/File Migration** tab and then **Create Job**.

Figure 4-87 Creating a job for migrating data from MySQL to Hive



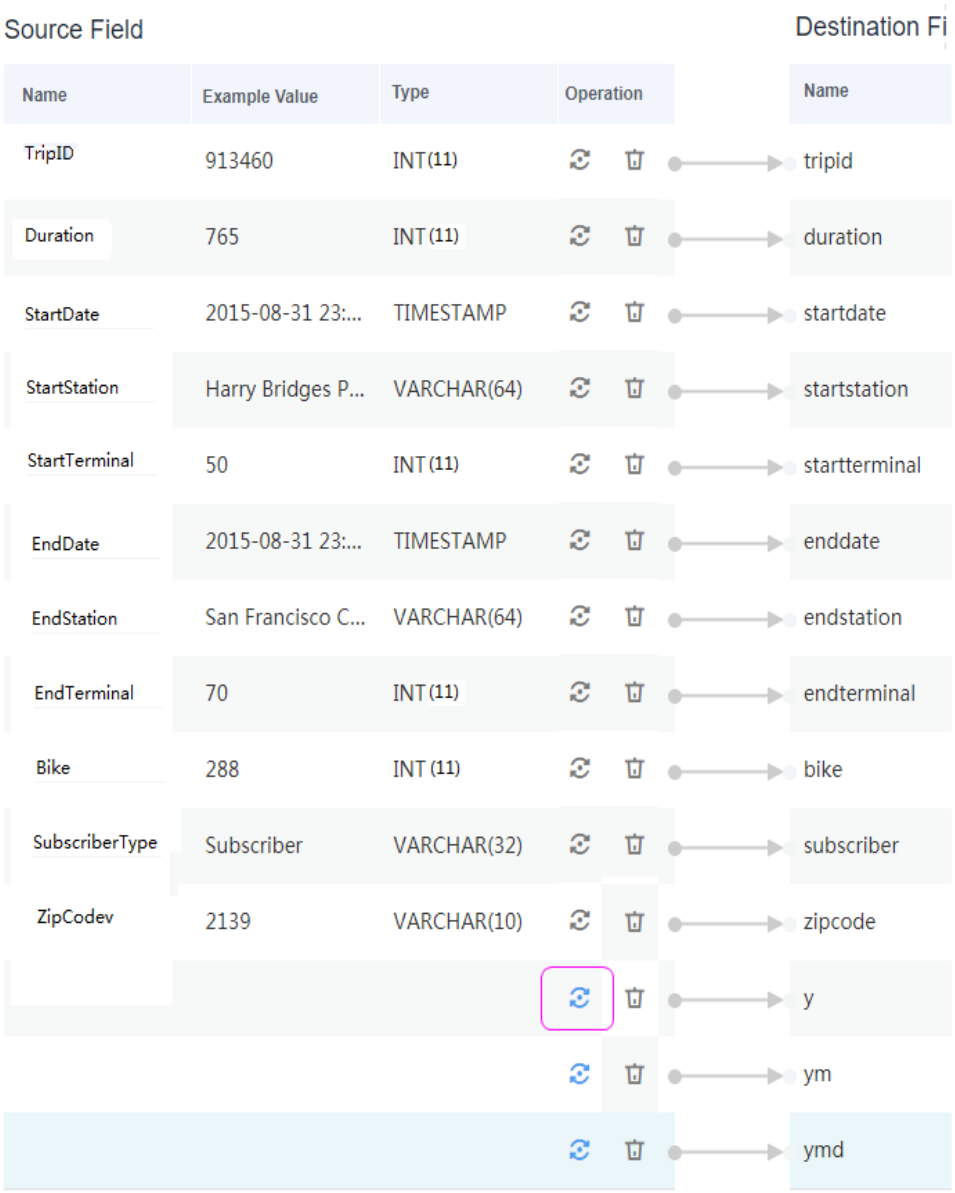
NOTE

Set **Clear Data Before Import** to **Yes**, so that the data in the Hive table will be cleared before data import.

Step 2 After configuring the parameters, click **Next** to go to the **Map Field** page shown in **Figure 4-88**.

Map the fields of the MySQL table and Hive table. The Hive table has three more fields **y**, **ym**, and **ymd** than the MySQL table, which are the Hive partition fields. Because the fields of the source table cannot be directly mapped to the destination table, you need to configure an expression to extract data from the **StartDate** field in the source table.

Figure 4-88 Hive field mapping



Step 3 Click to display the **Converter List** dialog box, and then choose **Create Converter** > **Expression conversion**. See **Figure 4-89**.

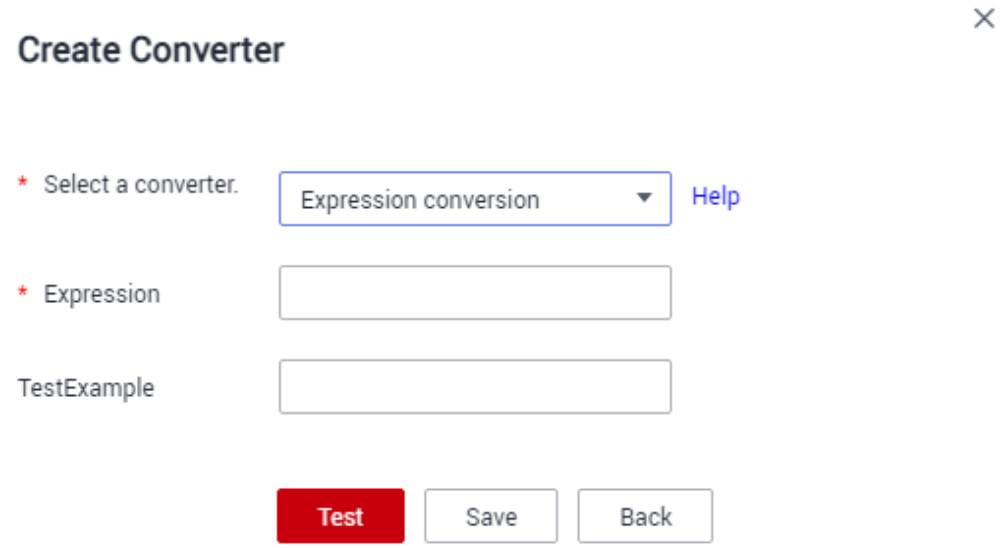
The expressions for the **y**, **ym**, and **ymd** fields are as follows:

DateUtils.format(DateUtils.parseDate(row[2],"yyyy-MM-dd HH:mm:ss.SSS"),"yyyy")

DateUtils.format(DateUtils.parseDate(row[2],"yyyy-MM-dd HH:mm:ss.SSS"),"yyyyMM")

DateUtils.format(DateUtils.parseDate(row[2],"yyyy-MM-dd HH:mm:ss.SSS"),"yyyyMMdd")

Figure 4-89 Configuring the expression



NOTE

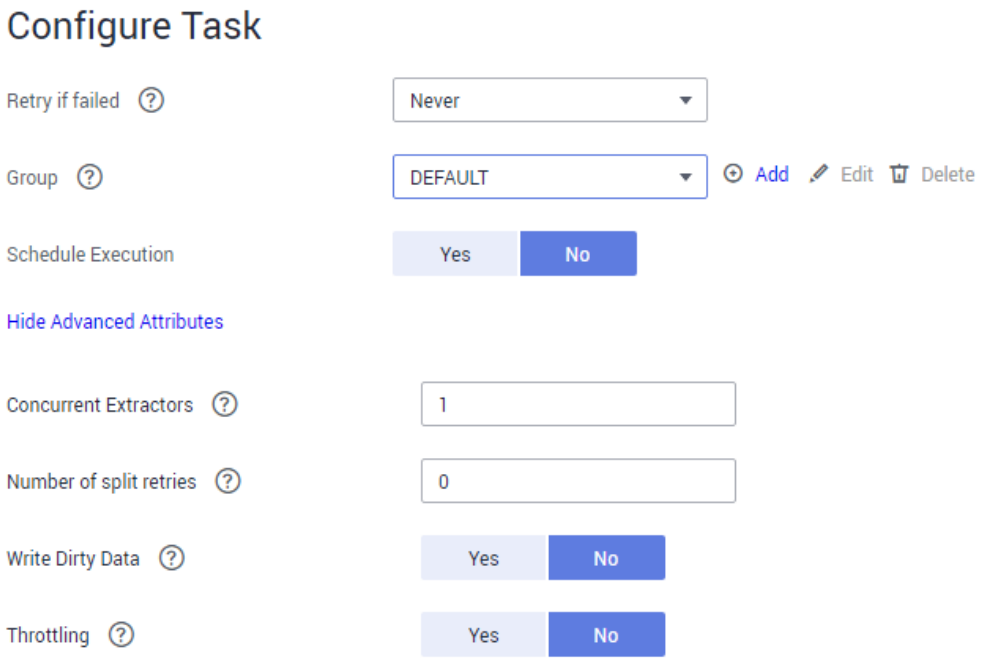
The expressions in CDM support field conversion of common character strings, dates, and values. For details, see **Converting Fields**.

Step 4 Click **Next** and set task parameters. Generally, retain the default values of all parameters.

In this step, you can configure the following optional functions:

- **Retry If Failed**: Determine whether to automatically retry the job if it fails. Retain the default value **Never**.
- **Group**: Select the group to which the job belongs. The default group is **DEFAULT**. On the **Job Management** page, jobs can be displayed, started, or exported by group.
- **Schedule Execution**: Determine whether to automatically execute the job at a scheduled time. Retain the default value **No** in this example.
- **Concurrent Extractors**: Enter the number of concurrent extractors. An appropriate value improves migration efficiency. For details, see **Performance Tuning**. Retain the default value **1**.
- **Write Dirty Data**: Specify this parameter if data that fails to be processed or filtered out during job execution needs to be written to OBS for future viewing. Before writing dirty data, create an OBS link on the CDM console. Retain the default value **No** so that dirty data is not recorded.

Figure 4-90 Configuring the task



Step 5 Click **Save and Run**. The **Job Management** page is displayed, on which you can view the job execution progress and result.

Step 6 After the job is successfully executed, in the **Operation** column of the job, click **Historical Record** to view the job's historical execution records and read/write statistics.

On the **Historical Record** page, click **Log** to view the job logs.

----End

4.10.4 Migrating Data from MySQL to OBS

Scenario

CDM supports table-to-OBS data migration. This section describes how to migrate tables from a MySQL database to OBS. The process is as follows:

1. **Creating a CDM Cluster and Binding an EIP to the Cluster**
2. **Creating a MySQL Link**
3. **Creating an OBS Link**
4. **Creating a Migration Job**

Prerequisites

- You have obtained the domain name, port number, AK, and SK for accessing OBS.
- You have obtained the IP address, port, database name, username, and password for connecting to the MySQL database. In addition, the user must have the read and write permissions on the MySQL database.

- You have uploaded the MySQL database driver on the **Job Management** > **Links** > **Driver Management** page.

Creating a CDM Cluster and Binding an EIP to the Cluster

Step 1 If CDM is used an independent service, create a CDM cluster by following the instructions in **Creating a CDM Cluster**. If CDM is used as a module of DataArts Studio, create a CDM cluster by following the instructions in **Creating a CDM Cluster**.

The key configurations are as follows:

The flavor of the CDM cluster is selected based on the amount of data to be migrated. Generally, cdm.medium meets the requirements for most migration scenarios.

Step 2 After the CDM cluster is created, on the **Cluster Management** page, click **Bind EIP** in the **Operation** column to bind an EIP to the cluster. The CDM cluster uses the EIP to access MySQL.

NOTE

If SSL encryption is configured for the access channel of a local data source, CDM cannot connect to the data source using the EIP.

----End

Creating a MySQL Link

Step 1 On the **Cluster Management** page, locate a cluster and click **Job Management** in the **Operation** column. On the displayed page, click the **Links** tab and then **Create Link**.

Figure 4-91 Selecting a connector

Step 2 Select **RDS for MySQL** and click **Next** to set the link parameters.

Click **Show Advanced Attributes** to view more optional parameters. For details, see **Link to an RDS for MySQL/MySQL Database**. Retain the default values of the optional parameters and configure the mandatory parameters according to **Table 4-126**.

Table 4-126 MySQL link parameters

Parameter	Description	Example Value
Name	Unique link name	mysqllink
Database Server	IP address or domain name of the MySQL database server	-
Port	MySQL database port	3306
Database Name	Name of the MySQL database	sqoop
Username	User who has the read, write, and delete permissions on the MySQL database	admin
Password	Password of the user	-
Use Local API	Whether to use the local API of the database for acceleration. (The system attempts to enable the **local_infile** system variable of the MySQL database.)	Yes
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-
local_infile Character Set	When using local_infile to import data to MySQL, you can configure the encoding format.	utf8
Driver Version	Before connecting CDM to a relational database, you need to upload the JDK 8 .jar driver of the relational database. Download the MySQL driver 5.1.48 from **https://downloads.mysql.com/archives/c-j/**, obtain **mysql-connector-java-5.1.48.jar**, and upload it.	-

Step 3 Click **Save**. The **Link Management** page is displayed.

NOTE

If an error occurs during the saving, the security settings of the MySQL database are incorrect. In this case, you need to enable the EIP of the CDM cluster to access the MySQL database.

----End

Creating an OBS Link

Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 4-92 Selecting a connector type

Step 2 Select **Object Storage Service (OBS)** and click **Next** to configure parameters for the OBS link.

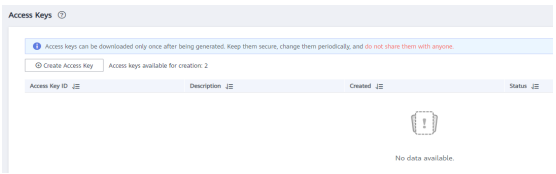
- **Name**: Enter a custom link name, for example, **obslink**.
- **OBS Server** and **Port**: Enter the actual OBS address information.
- **AK** and **SK**: Enter the AK and SK used for logging in to OBS.

  To obtain an access key, perform the following steps:

  a. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list.

  b. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 4-93**.

  Figure 4-93 Clicking Create Access Key

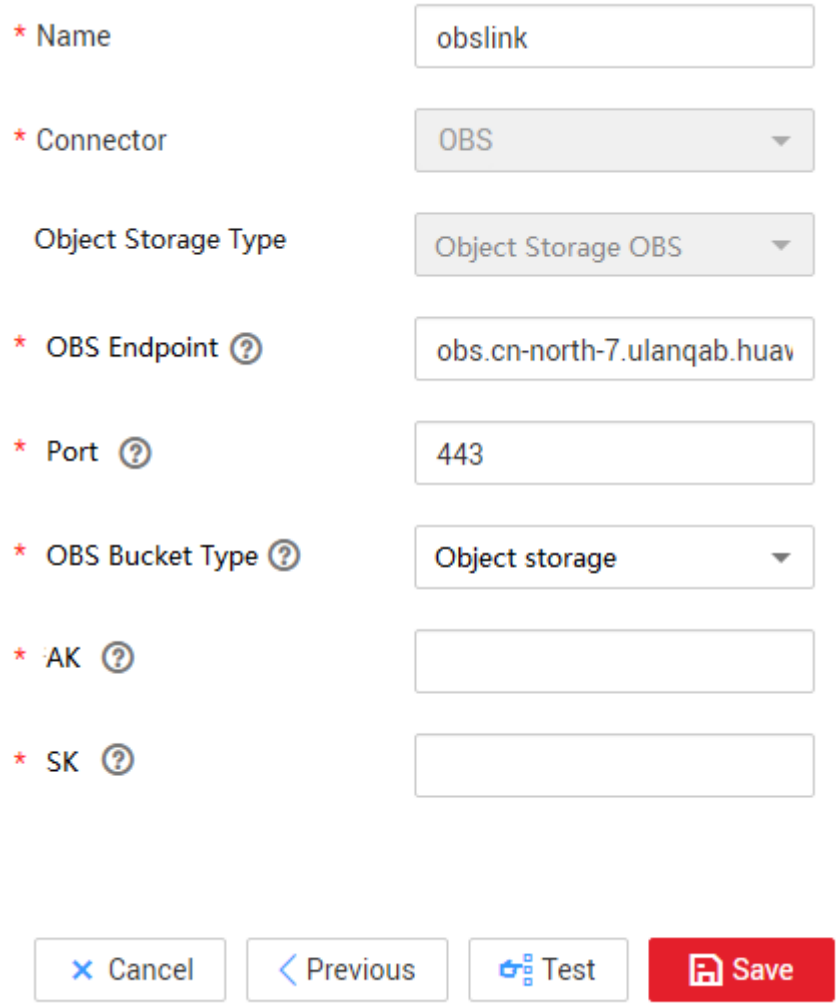


c. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**.

NOTE

- Only two access keys can be added for each user.
- To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.

Figure 4-94 Creating an OBS link



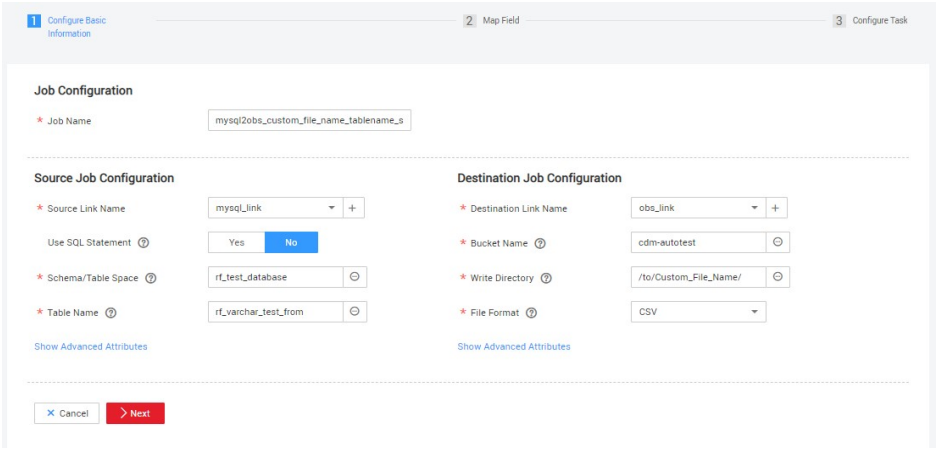
Step 3 Click **Save**. The **Link Management** page is displayed.

----End

Creating a Migration Job

Step 1 Choose **Table/File Migration** > **Create Job** to create a job for exporting data from the MySQL database to OBS.

Figure 4-95 Creating a job for migrating data from MySQL to OBS

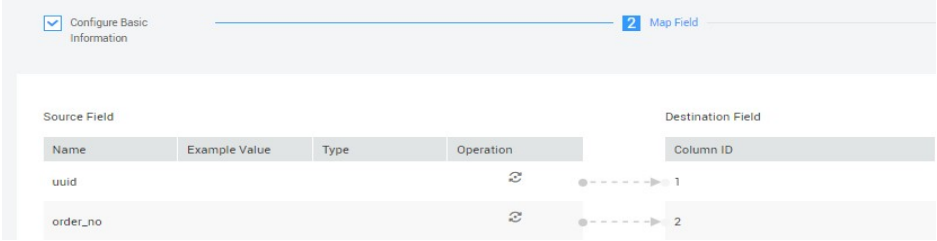


- **Job Name**: Enter a unique name.
- **Source Job Configuration**
  - **Source Link Name**: Select the **mysqllink** created in **Creating a MySQL Link**.
  - **Use SQL Statement**: Select **No**.
  - **Schema/Tablespace**: name of the schema or tablespace from which data is to be extracted
  - **Table Name**: name of the table from which data is to be extracted
  - Retain the default values of other optional parameters.
- **Destination Job Configuration**
  - **Destination Link Name**: Select the **obslink** created in **Creating an OBS Link**.
  - **Bucket Name**: Select the bucket from which the data will be migrated.
  - **Write Directory**: Enter the directory to which data is to be written on the OBS server.
  - **File Format**: Select **CSV**.
  - Retain the default values of the optional parameters in **Show Advanced Attributes**.

Step 2 Click **Next**. The **Map Field** page is displayed. CDM automatically matches the source and destination fields, as shown in **Figure 4-96**.

- If the field mapping is incorrect, you can drag the fields to adjust the mapping.
- The expressions in CDM support field conversion of common character strings, dates, and values. For details, see **Converting Fields**.

Figure 4-96 Table-to-file field mapping



Step 3 Click **Next** and set task parameters. Generally, retain the default values of all parameters.

In this step, you can configure the following optional functions:

- **Retry Upon Failure**: If the job fails to be executed, you can determine whether to automatically retry. Retain the default value **Never**.
- **Group**: Select the group to which the job belongs. The default group is **DEFAULT**. On the **Job Management** page, jobs can be displayed, started, or exported by group.
- **Schedule Execution**: Enable it if you need to configure scheduled jobs. Retain the default value **No**.
- **Concurrent Extractors**: Enter the number of extractors to be concurrently executed. CDM supports concurrent extraction of MySQL data. If indexes are configured for the source table, you can increase the number of concurrent extractors to accelerate the migration.
- **Write Dirty Data**: Specify this parameter if data that fails to be processed or filtered out during job execution needs to be written to OBS for future viewing. Before writing dirty data, create an OBS link. For file-to-table data migration, you are advised to write dirty data.
- **Delete Job After Completion**: Retain the default value **Do not delete**. You can also set this parameter to **Delete** to prevent an accumulation of too many migration jobs.

Step 4 Click **Save and Run**. The **Job Management** page is displayed, on which you can view the job execution progress and result.

Step 5 After the job is successfully executed, in the **Operation** column of the job, click **Historical Record** to view the job's historical execution records and read/write statistics.

On the **Historical Record** page, click **Log** to view the job logs.

----End

4.10.5 Migrating Data from MySQL to DWS

Scenario

CDM supports table-to-table data migration. This section describes how to migrate data from MySQL to DWS. The process is as follows:

1. **Creating a CDM Cluster and Binding an EIP to the Cluster**
2. **Creating a MySQL Link**
3. **Creating a DWS Link**
4. **Creating a Migration Job**

Prerequisites

- You have obtained the IP address, port number, database name, username, and password for connecting to DWS. In addition, you must have the read, write, and delete permissions on the DWS database.

- You have obtained the IP address, port, database name, username, and password for connecting to the MySQL database. In addition, the user must have the read and write permissions on the MySQL database.
- You have uploaded the MySQL database driver on the **Job Management** > **Links** > **Driver Management** page.

Creating a CDM Cluster and Binding an EIP to the Cluster

Step 1 If CDM is used an independent service, create a CDM cluster by following the instructions in **Creating a CDM Cluster**. If CDM is used as a module of DataArts Studio, create a CDM cluster by following the instructions in **Creating a CDM Cluster**.

The key configurations are as follows:

- The flavor of the CDM cluster is selected based on the amount of data to be migrated. Generally, cdm.medium meets the requirements for most migration scenarios.
- The VPC, subnet, and security group of the CDM cluster must be the same as those of the DWS cluster.

Step 2 After the CDM cluster is created, on the **Cluster Management** page, click **Bind EIP** in the **Operation** column to bind an EIP to the cluster. The CDM cluster uses the EIP to access MySQL.

NOTE

If SSL encryption is configured for the access channel of a local data source, CDM cannot connect to the data source using the EIP.

----End

Creating a MySQL Link

Step 1 On the **Cluster Management** page, locate a cluster and click **Job Management** in the **Operation** column. On the displayed page, click the **Links** tab and then **Create Link**.

Figure 4-97 Selecting a connector

Step 2 Select **RDS for MySQL** and click **Next** to set the link parameters.

Click **Show Advanced Attributes** to view more optional parameters. For details, see **Link to an RDS for MySQL/MySQL Database**. Retain the default values of the optional parameters and configure the mandatory parameters according to **Table 4-127**.

Table 4-127 MySQL link parameters

Parameter	Description	Example Value
Name	Unique link name	mysqllink
Database Server	IP address or domain name of the MySQL database server	-
Port	MySQL database port	3306
Database Name	Name of the MySQL database	sqoop
Username	User who has the read, write, and delete permissions on the MySQL database	admin
Password	Password of the user	-
Use Local API	Whether to use the local API of the database for acceleration. (The system attempts to enable the **local_infile** system variable of the MySQL database.)	Yes

Parameter	Description	Example Value
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-
local_infile Character Set	When using local_infile to import data to MySQL, you can configure the encoding format.	utf8
Driver Version	Before connecting CDM to a relational database, you need to upload the JDK 8 .jar driver of the relational database. Download the MySQL driver 5.1.48 from **https://downloads.mysql.com/archives/c-j/**, obtain **mysql-connector-java-5.1.48.jar**, and upload it.	-

Step 3 Click **Save**. The **Link Management** page is displayed.

NOTE

If an error occurs during the saving, the security settings of the MySQL database are incorrect. In this case, you need to enable the EIP of the CDM cluster to access the MySQL database.

----End

Creating a DWS Link

Step 1 On the **Cluster Management** page, locate a cluster and click **Job Management** in the **Operation** column. On the displayed page, click the **Links** tab and then **Create Link**.

Figure 4-98 Selecting a connector type

Step 2 Select **Data Warehouse Service** and click **Next** to configure the DWS link parameters. Set the mandatory parameters listed in **Table 4-128** and retain the default values for the optional parameters.

Table 4-128 DWS link parameters

Parameter	Description	Example Value
Name	Enter a unique link name.	dwslink
Database Server	IP address or domain name of the DWS database	192.168.0.3
Port	DWS database port	8000
Database Name	Name of the DWS database	db_demo
Username	User who has the read, write, and delete permissions on the DWS database	dbadmin
Password	Password of the user	-
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-
Agent	The agent function will be unavailable soon and does not need to be configured.	-
Import Mode	**COPY**: Migrate the source data to the DWS management node and then copy the data to DataNodes. To access DWS through the Internet, select **COPY**.	COPY

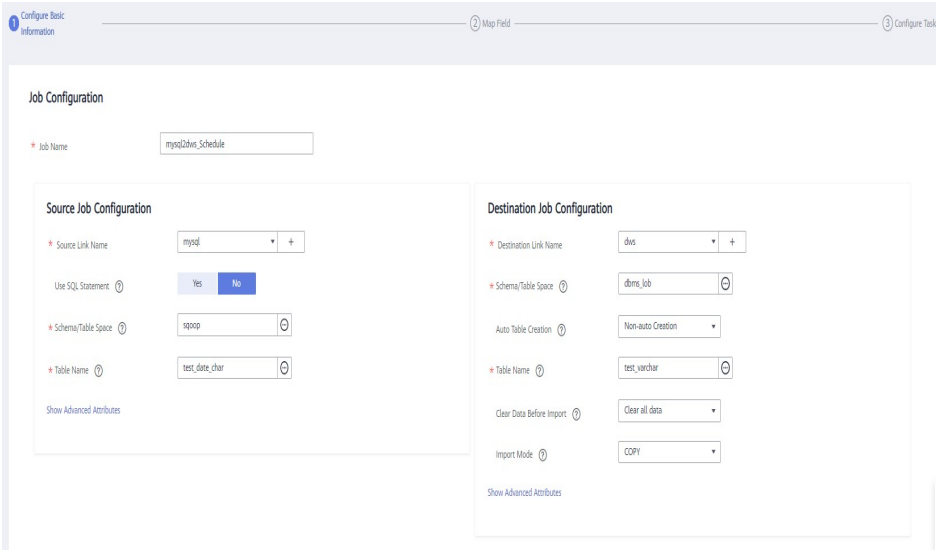
Step 3 Click **Save**.

----End

Creating a Migration Job

Step 1 Choose **Table/File Migration** > **Create Job** to create a job for exporting data from the MySQL database to DWS.

Figure 4-99 Creating a job for migrating data from MySQL to DWS



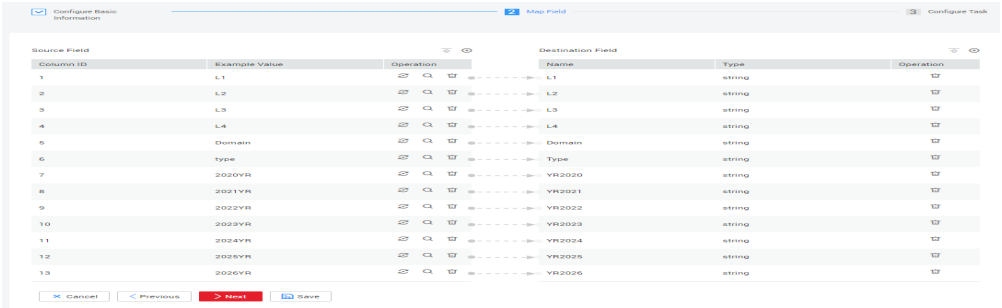
- **Job Name**: Enter a unique name.
- **Source Job Configuration**
  - **Source Link Name**: Select the **mysqllink** created in **Creating a MySQL Link**.
  - **Use SQL Statement**: Select **No**.
  - **Schema/Tablespace**: name of the schema or tablespace from which data is to be extracted
  - **Table Name**: name of the table from which data is to be extracted
  - Retain the default values of other optional parameters.
- **Destination Job Configuration**
  - **Destination Link Name**: Select the **dwslink** created in **Creating a DWS Link**.
  - **Schema/Tablespace**: Select the DWS database to which data is to be written.
  - **Auto Table Creation**: This parameter is displayed only when both the migration source and destination are relational databases.
  - **Table Name**: Name of the table to which data is to be written. You can enter a table name that does not exist. CDM automatically creates the table in DWS.
  - **isCompress**: whether to compress data. If you select **Yes**, high-level compression will be performed. CDM applies to compression scenarios where the I/O read/write volume is large and the CPU is sufficient (the computing load is relatively low). For more compression levels, see **Compression Levels**.
  - **Orientation**: You can create row- or column-store tables as needed. Generally, if a table contains many columns (called a wide table) and its query involves only a few columns, column storage is recommended. If a table contains only a few columns and a query includes most of the fields, row storage is recommended.

  - **Extend char length**: If the data encoding formats of the migration source and destination are different, the character length of the automatic table creation may be insufficient. If you select **Yes** for this parameter, the character length will be increased by three times during automatic table creation.
  - **Clear Data Before Import**: whether to clear data in the destination table before the migration task starts.

Step 2 Click **Next**. The **Map Field** page is displayed. CDM automatically matches the source and destination fields, as shown in **Figure 4-100**.

- If the field mapping is incorrect, you can drag the fields to adjust the mapping.
- The expressions in CDM support field conversion of common character strings, dates, and values. For details, see **Converting Fields**.

Figure 4-100 Table-to-table field mapping



Step 3 Click **Next** and set task parameters. Generally, retain the default values of all parameters.

In this step, you can configure the following optional functions:

- **Retry Upon Failure**: If the job fails to be executed, you can determine whether to automatically retry. Retain the default value **Never**.
- **Group**: Select the group to which the job belongs. The default group is **DEFAULT**. On the **Job Management** page, jobs can be displayed, started, or exported by group.
- **Schedule Execution**: Enable it if you need to configure scheduled jobs. Retain the default value **No**.
- **Concurrent Extractors**: Enter the number of extractors to be concurrently executed. You can increase the value of this parameter to improve migration efficiency.
- **Write Dirty Data**: Dirty data may be generated during data migration between tables. You are advised to select **Yes**.
- **Delete Job After Completion**: Retain the default value **Do not delete**.

Step 4 Click **Save and Run**. The **Job Management** page is displayed, on which you can view the job execution progress and result.

Step 5 After the job is successfully executed, in the **Operation** column of the job, click **Historical Record** to view the job's historical execution records and read/write statistics.

On the **Historical Record** page, click **Log** to view the job logs.

----End

4.10.6 Migrating an Entire MySQL Database to RDS

Scenario

This section describes how to migrate the entire on-premises MySQL database to RDS using the CDM's entire DB migration function.

Currently, CDM can migrate the entire on-premises MySQL database to RDS for MySQL, RDS for PostgreSQL, or RDS for SQL Server. The following describes how to migrate the entire database to RDS. The procedure is as follows:

1. **Creating a CDM Cluster and Binding an EIP to the Cluster**
2. **Creating a MySQL Link**
3. **Creating an RDS Link**
4. **Creating an Entire DB Migration Job**

Prerequisites

- You have sufficient EIP quota.
- You have obtained an RDS database instance and the database engine of this instance is MySQL.
- The on-premises MySQL database can be accessed through the public network. If the MySQL database is deployed on an on-premises data center or a third-party cloud, ensure that an IP address that can be accessed from the public network has been configured for the MySQL database, or the VPN or Direct Connect between the on-premises data center and the cloud service platform has been established.
- You have obtained the IP addresses, names, usernames, and passwords of the on-premises MySQL database and RDS for MySQL.
- You have uploaded the MySQL database driver on the **Job Management** > **Links** > **Driver Management** page.

Creating a CDM Cluster and Binding an EIP to the Cluster

Step 1 If CDM is used an independent service, create a CDM cluster by following the instructions in **Creating a CDM Cluster**. If CDM is used as a module of DataArts Studio, create a CDM cluster by following the instructions in **Creating a CDM Cluster**.

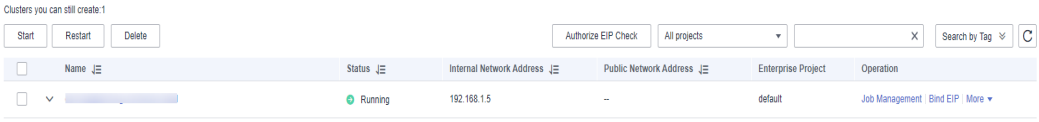
The key configurations are as follows:

- The flavor of the CDM cluster is selected based on the amount of data to be migrated. Generally, cdm.medium meets the requirements for most migration scenarios.
- The CDM cluster and the RDS for MySQL instance must be in the same VPC. In addition, it is recommended that the CDM cluster be in the same subnet and security group as the RDS for MySQL instance.

- If the same subnet and security group cannot be used for security purposes, ensure that a security group rule has been configured to allow the CDM cluster to access the RDS for MySQL instance.

Step 2 After the CDM cluster is created, on the **Cluster Management** page, click **Bind EIP** in the **Operation** column to bind an EIP to the cluster. The CDM cluster uses the EIP to access the on-premises MySQL database.

Figure 4-101 Cluster list



NOTE

If SSL encryption is configured for the access channel of a local data source, CDM cannot connect to the data source using the EIP.

----End

Creating a MySQL Link

Step 1 On the **Cluster Management** page, locate a cluster and click **Job Management** in the **Operation** column. On the displayed page, click the **Links** tab and then **Create Link**.

Figure 4-102 Selecting a connector

Step 2 Select **RDS for MySQL** and click **Next** to set the link parameters.

Click **Show Advanced Attributes** to view more optional parameters. For details, see **Link to an RDS for MySQL/MySQL Database**. Retain the default values of

the optional parameters and configure the mandatory parameters according to **Table 4-129**.

Table 4-129 MySQL link parameters

Parameter	Description	Example Value
Name	Unique link name	mysqllink
Database Server	IP address or domain name of the MySQL database server	-
Port	MySQL database port	3306
Database Name	Name of the MySQL database	sqoop
Username	User who has the read, write, and delete permissions on the MySQL database	admin
Password	Password of the user	-
Use Local API	Whether to use the local API of the database for acceleration. (The system attempts to enable the **local_infile** system variable of the MySQL database.)	Yes
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-
local_infile Character Set	When using local_infile to import data to MySQL, you can configure the encoding format.	utf8
Driver Version	Before connecting CDM to a relational database, you need to upload the JDK 8 .jar driver of the relational database. Download the MySQL driver 5.1.48 from **https://downloads.mysql.com/archives/c-j/**, obtain **mysql-connector-java-5.1.48.jar**, and upload it.	-

Step 3 Click **Save**. The **Link Management** page is displayed.

NOTE

If an error occurs during the saving, the security settings of the MySQL database are incorrect. In this case, you need to enable the EIP of the CDM cluster to access the MySQL database.

----End

Creating an RDS Link

Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 4-103 Selecting a connector type

Step 2 Select **RDS for MySQL** and click **Next** to configure parameters for the RDS for MySQL link.

- **Name**: Enter a custom link name, for example, **rds_link**.
- **Database Server** and **Port**: Enter the address information about the RDS for MySQL database.
- **Database Name**: Enter the name of the RDS for MySQL database.
- **Username** and **Password**: Enter the username and password used for logging in to the database.

NOTE

- During RDS link creation, if **Use Local API** in **Show Advanced Attributes** is set to **Yes**, you can use the LOAD DATA function provided by MySQL to speed up data import.
- The LOAD DATA function is disabled by default on RDS for MySQL, so you need to modify the parameter group of the MySQL instance and set **local_infile** to **ON** to enable this function.
- If the **local_infile** parameter group cannot be edited, it is the default parameter group. You need to create a parameter group and modify its value, and apply it to the MySQL instance of RDS.

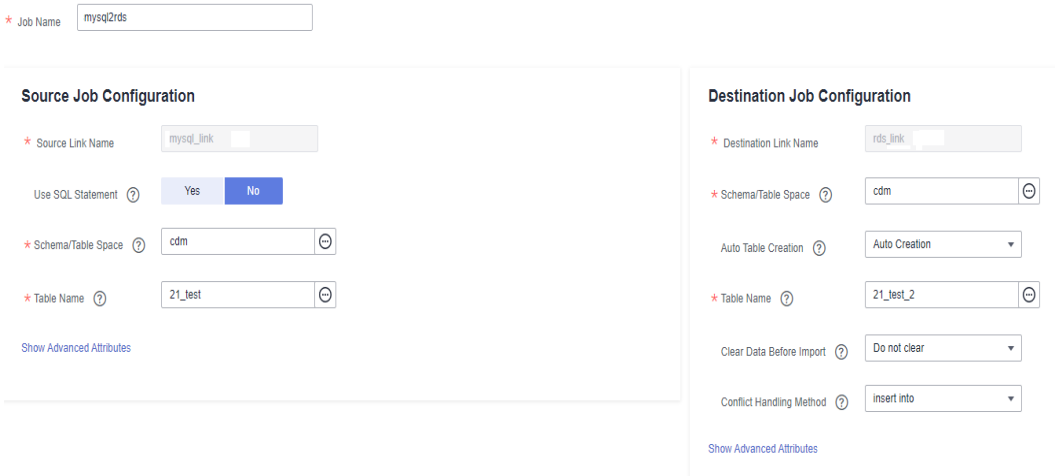
Step 3 Click **Save**. The **Link Management** page is displayed.

----End

Creating an Entire DB Migration Job

Step 1 After the two links are created, choose **Entire DB Migration** > **Create Job** to create a migration job. See **Figure 4-104**.

Figure 4-104 Creating an entire DB migration job



- **Job Name**: Enter a name for the entire DB migration job.
- **Source Job Configuration**
  - **Source Link Name**: Select the **mysqllink** created in **Creating a MySQL Link**.
  - **Schema/Tablespace**: Select the on-premises MySQL database from which data is to be exported.
- **Destination Job Configuration**
  - **Destination Link Name**: Select the **rds_link** link created in **Creating an RDS Link**.
  - **Schema/Tablespace**: Select the name of the RDS database to which data is to be imported.
  - **Auto Table Creation**: Select **Auto creation**, which indicates that CDM automatically creates tables in the RDS database when tables of the on-premises MySQL database do not exist in the RDS database.
  - **Clear Data Before Import**: Select **Yes**, which indicates that when a table with the same name as the table in the on-premises MySQL database exists in the RDS database, CDM clears data in the table on RDS.
  - **Constraint Conflict Handling**: Select **insert into**.
  - Retain the default values of the optional parameters in **Show Advanced Attributes**.

Step 2 Click **Next**. The page for selecting tables to be migrated is displayed. You can select all or part of tables to migrate.

Step 3 Click **Save and Run** and CDM immediately starts the entire DB migration job.

When the job starts running, a sub-job will be generated for each table. You can click the job name to view the sub-job list.

Step 4 In the **Operation** column of the job, click **Historical Record** to view the job's historical execution records and read/write statistics.

There are no logs for the entire DB migration job. However, the sub-jobs have logs. On the **Historical Record** page of the sub-jobs, click **Log** to view the job logs.

----End

4.10.7 Migrating Data from Oracle to CSS

Scenario

Cloud Search Service provides users with structured and unstructured data search, statistics, and report capabilities. This section describes how to use CDM to migrate data from the Oracle database to Cloud Search Service. The procedure is as follows:

1. **Creating a CDM Cluster and Binding an EIP to the Cluster**
2. **Creating a Cloud Search Service Link**
3. **Creating an Oracle Link**
4. **Creating a Migration Job**

Prerequisites

- You have subscribed to Cloud Search Service and obtained the IP address and port number of the Cloud Search Service cluster.
- You have obtained the IP address, name, username, and password of the Oracle database.
- If the Oracle database is deployed on an on-premises data center or a third-party cloud, ensure that an IP address that can be accessed from the public network has been configured for the Oracle database, or the VPN or Direct Connect between the on-premises data center and Huawei Cloud has been established.
- You have uploaded the Oracle database driver on the **Job Management** > **Links** > **Driver Management** page.

Creating a CDM Cluster and Binding an EIP to the Cluster

Step 1 If CDM is used an independent service, create a CDM cluster by following the instructions in **Creating a CDM Cluster**. If CDM is used as a module of DataArts Studio, create a CDM cluster by following the instructions in **Creating a CDM Cluster**.

The key configurations are as follows:

- The flavor of the CDM cluster is selected based on the amount of data to be migrated. Generally, cdm.medium meets the requirements for most migration scenarios.
- The CDM and Cloud Search Service clusters must be in the same VPC. In addition, it is recommended that the CDM cluster be in the same subnet and security group as the Cloud Search Service cluster.
- If the same subnet and security group cannot be used for security purposes, ensure that a security group rule has been configured to allow the CDM cluster to access the Cloud Search Service cluster.

Step 2 After the CDM cluster is created, on the **Cluster Management** page, click **Bind EIP** in the **Operation** column to bind an EIP to the cluster. The CDM cluster uses the EIP to access the Oracle data source.

NOTE

If SSL encryption is configured for the access channel of a local data source, CDM cannot connect to the data source using the EIP.

----End

Creating a Cloud Search Service Link

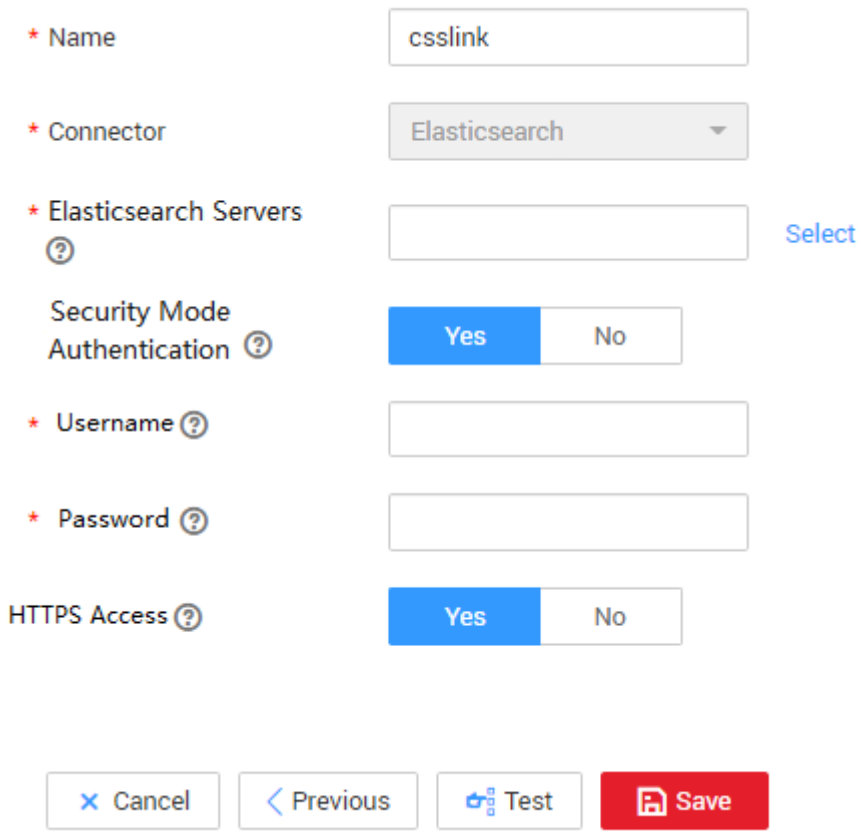
Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 4-105 Selecting a connector

Step 2 Select **Cloud Search Service** and click **Next**. On the page that is displayed, configure the CSS link parameters.

- **Name**: Enter a custom link name, for example, **csslink**.
- **Elasticsearch Server List**: Enter the IP address and port number of the Cloud Search Service cluster (cluster later than 5.*x*). The format is *ip:port*. Use semicolons to separate multiple addresses. For example, **192.168.0.1:9200;192.168.0.2:9200**.
- **Username** and **Password**: Enter the username and password used for logging in to the Cloud Search Service cluster. The user must have the read and write permissions on the database.

Figure 4-106 Creating a CSS link



Step 3 Click **Save**. The **Link Management** page is displayed.

----End

Creating an Oracle Link

Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 4-107 Selecting a connector type

Step 2 Select **Oracle** and click **Next** to configure parameters for the Oracle link.

- **Name**: Enter a custom link name, for example, **oracle_link**.
- **Database Server** and **Port**: Enter the address and port number of the Oracle server.
- **Database Name**: Enter the name of the Oracle database whose data is to be exported.
- **Username** and **Password**: Enter the username and password used for logging in to the Oracle database. The user must have the permission to read the Oracle metadata.

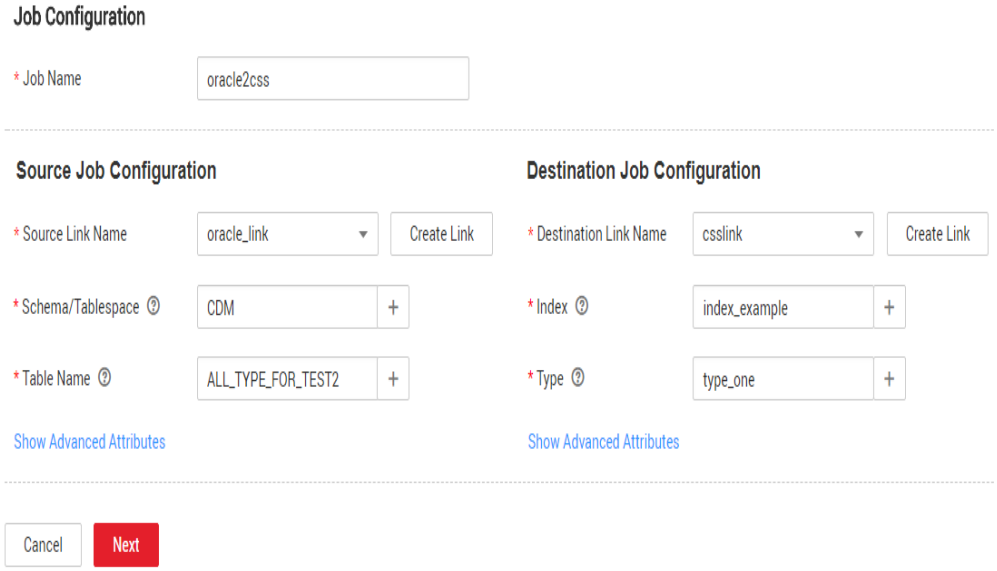
Step 3 Click **Save**. The **Link Management** page is displayed.

----End

Creating a Migration Job

Step 1 Choose **Table/File Migration** > **Create Job** to create a job for exporting data from the Oracle database to Cloud Search Service.

Figure 4-108 Creating a job for migrating data from Oracle to Cloud Search Service

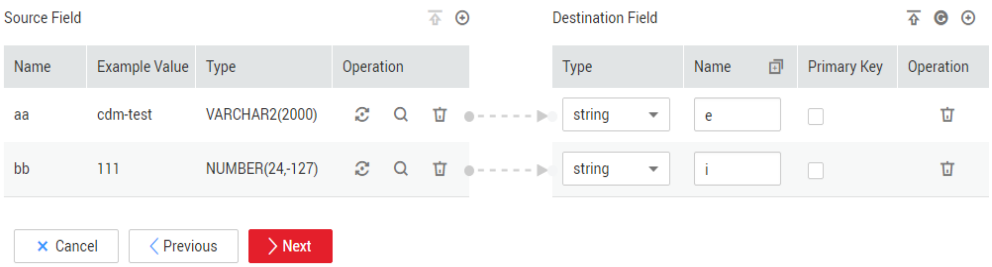


- **Job Name**: Enter a unique name.
- **Source Job Configuration**
  - **Source Link Name**: Select the **oracle_link** link created in **Creating an Oracle Link**.
  - **Schema/Tablespace**: Enter the name of the database whose data is to be migrated.
  - **Table Name**: Enter the name of the table to be migrated.
  - Retain the default values of the optional parameters in **Show Advanced Attributes**.
- **Destination Job Configuration**
  - **Destination Link Name**: Select the **csslink** link created in **Creating a Cloud Search Service Link**.
  - **Index**: Select the Elasticsearch index of the data to be written. You can also enter a new index. CDM automatically creates the index on Cloud Search Service.
  - **Type**: Select the Elasticsearch type of the data to be written. You can enter a new type. CDM automatically creates a type at the migration destination.
  - Retain the default values of the optional parameters in **Show Advanced Attributes**.

Step 2 Click **Next**. The **Map Field** page is displayed. CDM automatically matches the source and destination fields, as shown in **Figure 4-109**.

- If the field mapping is incorrect, you can drag the fields to adjust the mapping.
- If the type is automatically created at the migration destination, you need to configure the type and name of each field.
- CDM supports field conversion during the migration. For details, see **Converting Fields**.

Figure 4-109 Field mapping of Cloud Search Service

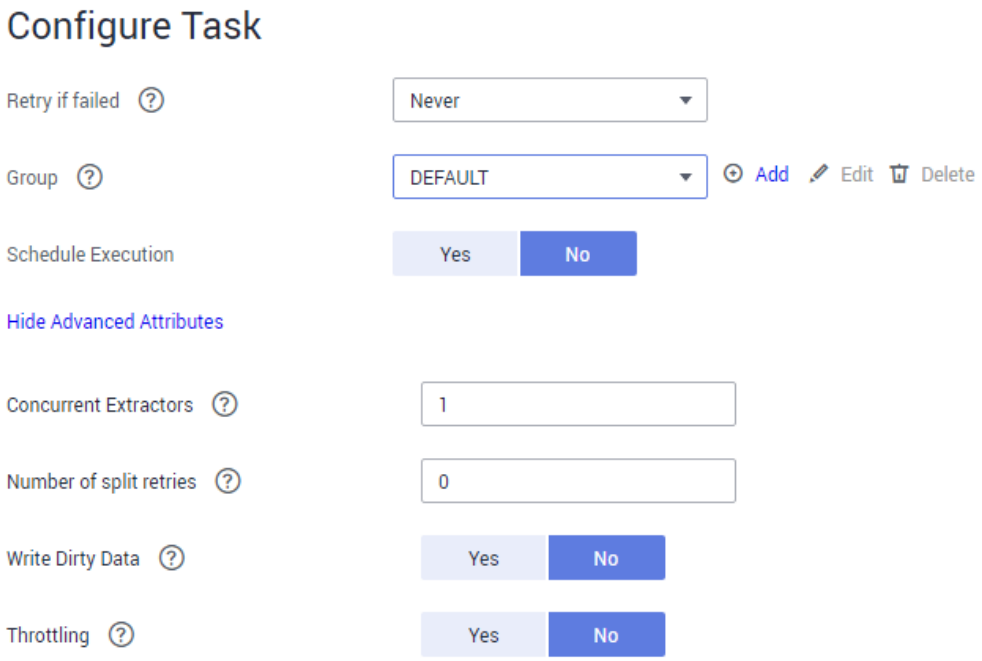


Step 3 Click **Next** and set task parameters. Generally, retain the default values of all parameters.

In this step, you can configure the following optional functions:

- **Retry If Failed**: Determine whether to automatically retry the job if it fails. Retain the default value **Never**.
- **Group**: Select the group to which the job belongs. The default group is **DEFAULT**. On the **Job Management** page, jobs can be displayed, started, or exported by group.
- **Schedule Execution**: Determine whether to automatically execute the job at a scheduled time. Retain the default value **No** in this example.
- **Concurrent Extractors**: Enter the number of concurrent extractors. An appropriate value improves migration efficiency. For details, see **Performance Tuning**. Retain the default value **1**.
- **Write Dirty Data**: Specify this parameter if data that fails to be processed or filtered out during job execution needs to be written to OBS for future viewing. Before writing dirty data, create an OBS link on the CDM console. Retain the default value **No** so that dirty data is not recorded.

Figure 4-110 Configuring the task



Step 4 Click **Save and Run**. The **Job Management** page is displayed, on which you can view the job execution progress and result.

Step 5 After the job is successfully executed, in the **Operation** column of the job, click **Historical Record** to view the job's historical execution records and read/write statistics.

On the **Historical Record** page, click **Log** to view the job logs.

----End

4.10.8 Migrating Data from Oracle to DWS

Scenario

CDM supports table-to-table migration. This section describes how to use CDM to migrate data from Oracle to Data Warehouse Service (DWS). The procedure is as follows:

1. **Creating a CDM Cluster and Binding an EIP to the Cluster**
2. **Creating an Oracle Link**
3. **Creating a DWS Link**
4. **Creating a Migration Job**

Prerequisites

- You have obtained a DWS cluster and the IP address, port number, database name, username, and password for connecting to the DWS database. In addition, you must have the read, write, and delete permissions on the DWS database.
- You have obtained the IP address, name, username, and password of the Oracle database.
- If the Oracle database is deployed on an on-premises data center or a third-party cloud, ensure that an IP address that can be accessed from the public network has been configured for the Oracle database, or the VPN or Direct Connect between the on-premises data center and Huawei Cloud has been established.
- You have uploaded the Oracle database driver on the **Job Management** > **Links** > **Driver Management** page.

Creating a CDM Cluster and Binding an EIP to the Cluster

Step 1 If CDM is used an independent service, create a CDM cluster by following the instructions in **Creating a CDM Cluster**. If CDM is used as a module of DataArts Studio, create a CDM cluster by following the instructions in **Creating a CDM Cluster**.

The key configurations are as follows:

- The flavor of the CDM cluster is selected based on the amount of data to be migrated. Generally, cdm.medium meets the requirements for most migration scenarios.

- The VPC, subnet, and security group of the CDM cluster must be the same as those of the DWS cluster.
- If the same subnet and security group cannot be used, for security reasons, ensure that a security group rule has been configured to allow the CDM cluster to access the CSS cluster.

Step 2 After the CDM cluster is created, locate the row that contains the cluster and click **Bind EIP** in the **Operation** column. (CDM uses an EIP to access the Oracle data source.)

NOTE

If SSL encryption is configured for the access channel of a local data source, CDM cannot connect to the data source using the EIP.

----End

Creating an Oracle Link

Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 4-111 Selecting a connector

Step 2 Select **Oracle** and click **Next** to configure parameters for the link.

Figure 4-112 Creating an Oracle link

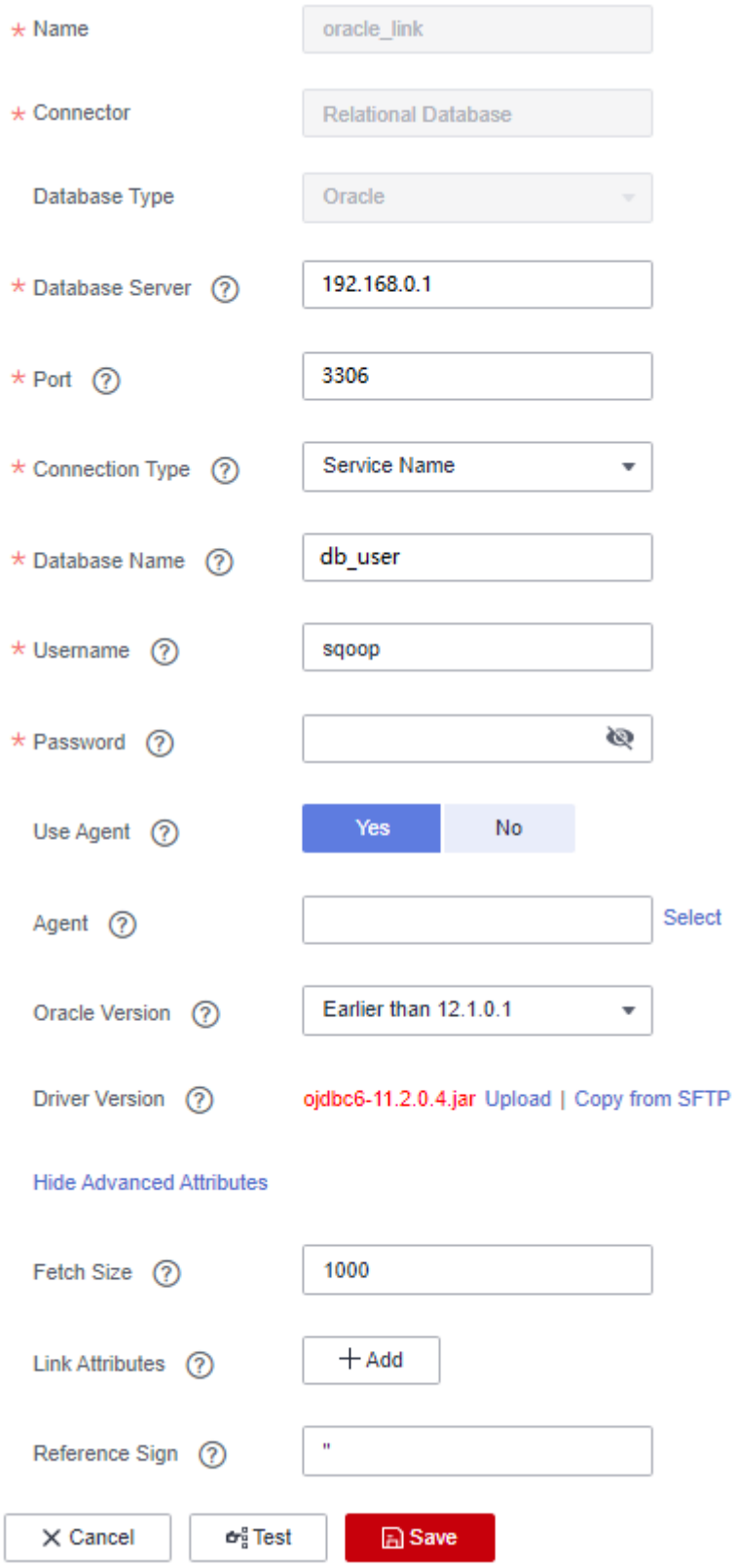


Table 4-130 Oracle link parameters

Parameter	Description	Example Value
Name	Enter a unique link name.	oracle_link
Database Server	Database server domain name or IP address	192.168.0.1
Port	Oracle database port	3306
Connection Type	Type of the Oracle database link	Service Name
Database Name	Name of the database to be connected	db_user
Username	User who has the read permission of the Oracle database	admin
Password	Password used for logging in to the Oracle database	-
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-
Agent	The agent function will be unavailable soon and does not need to be configured.	-
Oracle Version	The latest version is used by default. If the version is incompatible, select another version.	Later than 12.1
Driver Version	A driver version that adapts to the Oracle database	-
Fetch Size	Number of rows obtained by each request	1000
Link Attributes	Custom attributes of the link	useCompression=true
Reference Sign	Delimiter used to separate referenced table names or column names This parameter is left blank by default.	'

Step 3 Click **Save**. The **Links** page is displayed.

----End

Creating a DWS Link

Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 4-113 Selecting a connector type

Step 2 Select **Data Warehouse Service** and click **Next** to configure the DWS link parameters. Set the mandatory parameters listed in **Table 4-131** and retain the default values for the optional parameters.

Table 4-131 DWS link parameters

Parameter	Description	Example Value
Name	Enter a unique link name.	dwslink
Database Server	IP address or domain name of the DWS database	192.168.0.3
Port	DWS database port	8000
Database Name	Name of the DWS database	db_demo
Username	User who has the read, write, and delete permissions on the DWS database	dbadmin
Password	Password of the user	-
Use Agent	The agent function will be unavailable soon and does not need to be configured.	-
Agent	The agent function will be unavailable soon and does not need to be configured.	-

Parameter	Description	Example Value
Import Mode	**COPY**: Migrate the source data to the DWS management node and then copy the data to DataNodes. To access DWS through the Internet, select **COPY**.	COPY

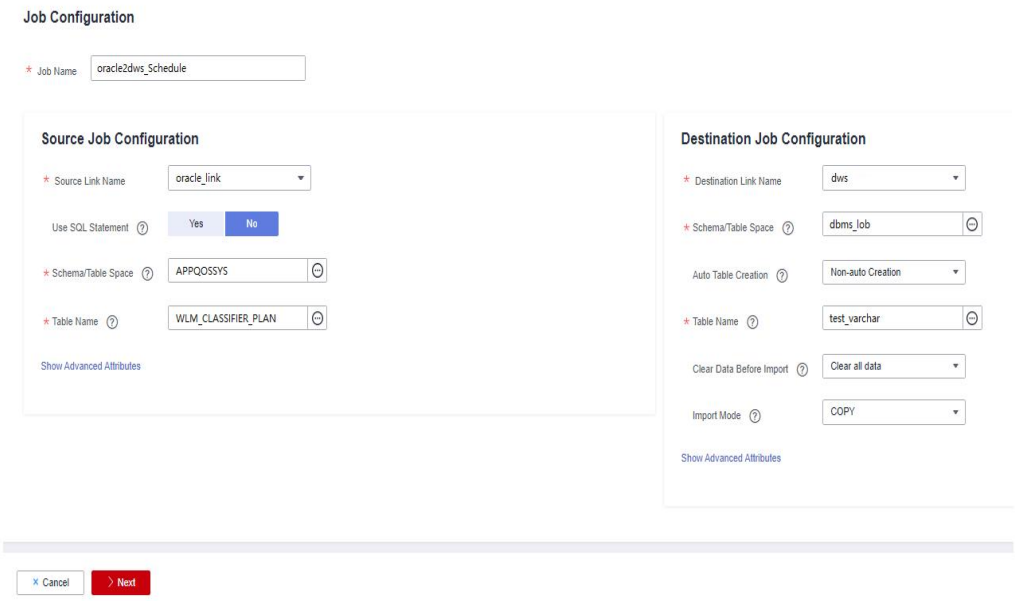
Step 3 Click **Save**.

----End

Creating a Migration Job

Step 1 Choose **Table/File Migration** > **Create Job** to create a job for exporting data from the Oracle database to DWS.

Figure 4-114 Creating a job for migrating data from Oracle to DWS



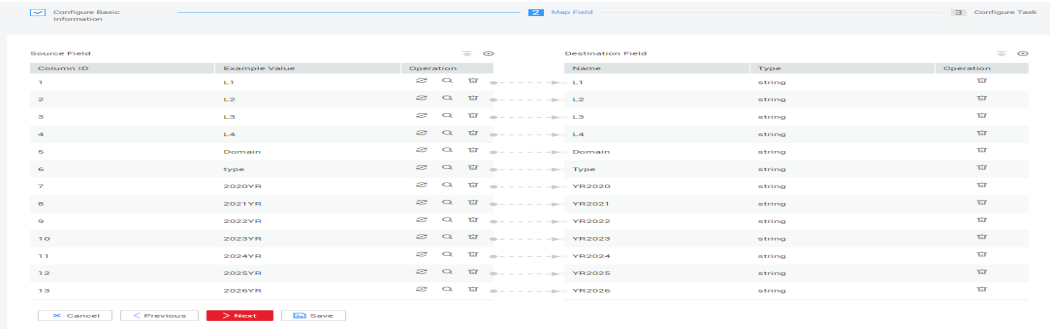
- **Job Name**: Enter a unique name.
- **Source Job Configuration**
  - **Source Link Name**: Select the **oracle_link** created in **Creating an Oracle Link**.
  - **Schema/Tablespace**: Enter the name of the database whose data is to be migrated.
  - **Table Name**: Enter the name of the table whose data is to be migrated.
  - Retain the default values of the optional parameters in **Show Advanced Attributes**.
- **Destination Job Configuration**
  - **Destination Link Name**: Select the **dwslink** created in **Creating a DWS Link**.

- **Schema/Tablespace**: Select the DWS database to which data is to be written.
- **Auto Table Creation**: This parameter is displayed only when both the migration source and destination are relational databases.
- **Table Name**: Name of the table to which data is to be written. You can enter a table name that does not exist. CDM automatically creates the table in DWS.
- **Orientation**: You can create row- or column-store tables as needed. Generally, if a table contains many columns (called a wide table) and its query involves only a few columns, column storage is recommended. If a table contains only a few columns and a query includes most of the fields, row storage is recommended.
- **Extend char length**: If the data encoding formats of the migration source and destination are different, the character length of the automatic table creation may be insufficient. If you select **Yes** for this parameter, the character length will be increased by three times during automatic table creation.
- **Clear Data Before Import**: whether to clear data in the destination table before the migration task starts.

Step 2 Click **Next**. The **Map Field** page is displayed. CDM automatically matches the source and destination fields, as shown in **Figure 4-115**.

- If the field mapping is incorrect, you can drag the fields to adjust the mapping.
- The expressions in CDM support field conversion of common character strings, dates, and values. For details, see **Converting Fields**.

Figure 4-115 Table-to-table field mapping



Step 3 Click **Next** and set task parameters. Generally, retain the default values of all parameters.

In this step, you can configure the following optional functions:

- **Retry Upon Failure**: If the job fails to be executed, you can determine whether to automatically retry. Retain the default value **Never**.
- **Group**: Select the group to which the job belongs. The default group is **DEFAULT**. On the **Job Management** page, jobs can be displayed, started, or exported by group.
- **Schedule Execution**: Enable it if you need to configure scheduled jobs. Retain the default value **No**.

- **Concurrent Extractors**: Enter the number of extractors to be concurrently executed. You can increase the value of this parameter to improve migration efficiency.
- **Write Dirty Data**: Dirty data may be generated during data migration between tables. You are advised to select **Yes**.
- **Delete Job After Completion**: Retain the default value **Do not delete**.

Step 4 Click **Save and Run**. The **Job Management** page is displayed, on which you can view the job execution progress and result.

Step 5 After the job is successfully executed, in the **Operation** column of the job, click **Historical Record** to view the job's historical execution records and read/write statistics.

On the **Historical Record** page, click **Log** to view the job logs.

----End

NOTE

If the migration times out because writing data to the destination costs a long time, reduce the value of the **Fetch Size** parameter.

4.10.9 Migrating Data from OBS to CSS

Scenario

CDM supports data migration between cloud services. This section describes how to use CDM to migrate data from OBS to CSS. The procedure is as follows:

1. **Creating a CDM Cluster**
2. **Creating a Cloud Search Service Link**
3. **Creating an OBS Link**
4. **Creating a Migration Job**

Prerequisites

- You have obtained the domain name, port number, AK, and SK for accessing OBS.
- You have subscribed to Cloud Search Service and obtained the IP address and port number of the Cloud Search Service cluster.

Creating a CDM Cluster

If CDM is used an independent service, create a CDM cluster by following the instructions in **Creating a CDM Cluster**. If CDM is used as a module of DataArts Studio, create a CDM cluster by following the instructions in **Creating a CDM Cluster**.

The key configurations are as follows:

- The flavor of the CDM cluster is selected based on the amount of data to be migrated. Generally, cdm.medium meets the requirements for most migration scenarios.

- The CDM and Cloud Search Service clusters must be in the same VPC. In addition, it is recommended that the CDM cluster be in the same subnet and security group as the Cloud Search Service cluster.
- If the same subnet and security group cannot be used for security purposes, ensure that a security group rule has been configured to allow the CDM cluster to access the Cloud Search Service cluster.

Creating a Cloud Search Service Link

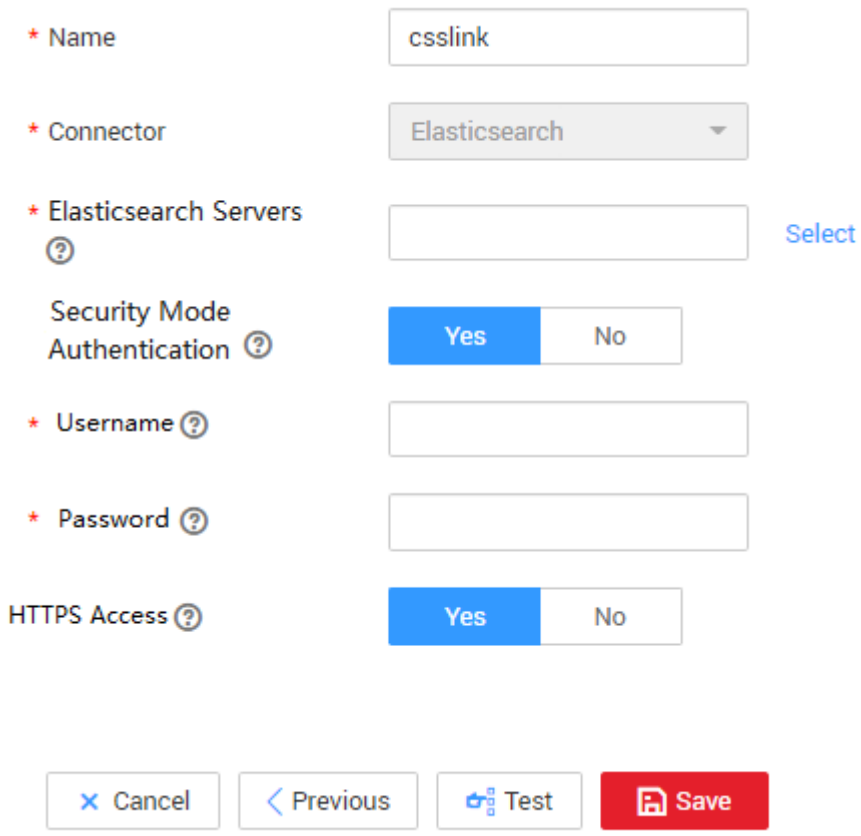
Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 4-116 Selecting a connector

Step 2 Select **Cloud Search Service** and click **Next**. On the page that is displayed, configure the CSS link parameters.

- **Name**: Enter a custom link name, for example, **csslink**.
- **Elasticsearch Server List**: Enter the IP address and port number of the Cloud Search Service cluster (cluster later than 5.*x*). The format is *ip:port*. Use semicolons to separate multiple addresses. For example, **192.168.0.1:9200;192.168.0.2:9200**.
- **Username** and **Password**: Enter the username and password used for logging in to the Cloud Search Service cluster. The user must have the read and write permissions on the database.

Figure 4-117 Creating a CSS link



Step 3 Click **Save**. The **Link Management** page is displayed.

----End

Creating an OBS Link

Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 4-118 Selecting a connector type

Step 2 Select **Object Storage Service (OBS)** and click **Next** to configure parameters for the OBS link.

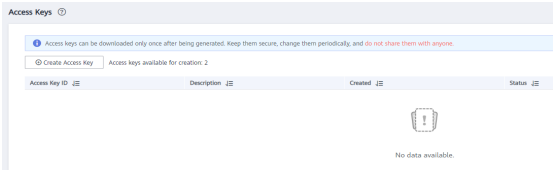
- **Name**: Enter a custom link name, for example, **obslink**.
- **OBS Server** and **Port**: Enter the actual OBS address information.
- **AK** and **SK**: Enter the AK and SK used for logging in to OBS.

  To obtain an access key, perform the following steps:

  a. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list.

  b. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 4-119**.

  Figure 4-119 Clicking Create Access Key

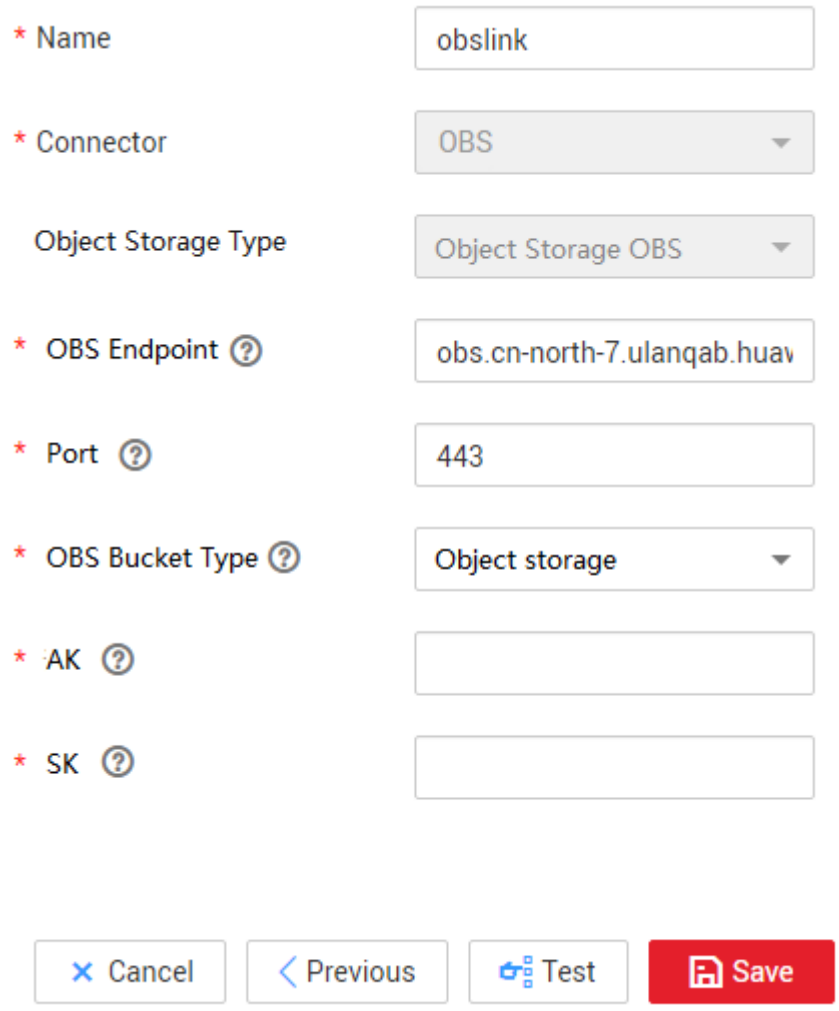
  

  c. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**.

  NOTE

  - Only two access keys can be added for each user.
  - To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.

Figure 4-120 Creating an OBS link



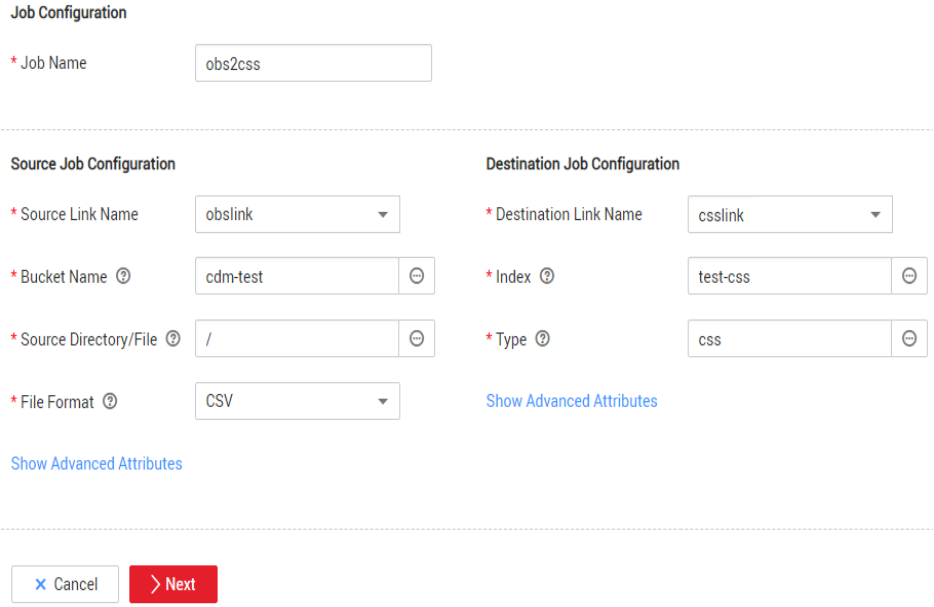
Step 3 Click **Save**. The **Link Management** page is displayed.

----End

Creating a Migration Job

Step 1 Choose **Table/File Migration** > **Create Job** to create a job for exporting data from OBS to Cloud Search Service.

Figure 4-121 Creating a job for migrating data from OBS to Cloud Search Service



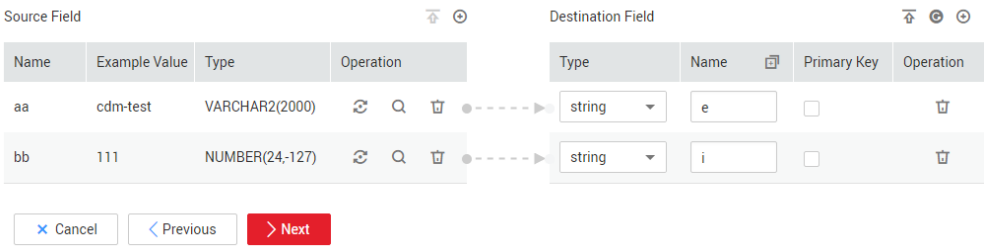
- **Job Name**: Enter a unique name.
- **Source Job Configuration**
  - **Source Link Name**: Select the **obslink** link created in **Creating an OBS Link**.
  - **Bucket Name**: Select the bucket from which the data will be migrated.
  - **Source Directory/File**: Set this parameter to the path of the data to be migrated. You can migrate all directories and files in the bucket.
  - **File Format**: Select **CSV** for migrating files to a data table.
  - Retain the default values of the optional parameters in **Show Advanced Attributes**.
- **Destination Job Configuration**
  - **Destination Link Name**: Select the **csslink** link created in **Creating a Cloud Search Service Link**.
  - **Index**: Select the Elasticsearch index of the data to be written. You can also enter a new index. CDM automatically creates the index on Cloud Search Service.
  - **Type**: Select the Elasticsearch type of the data to be written. You can enter a new type. CDM automatically creates a type at the migration destination.
  - Retain the default values of the optional parameters in **Show Advanced Attributes**.

Step 2 Click **Next**. The **Map Field** page is displayed. CDM automatically matches the source and destination fields. See **Figure 4-122**.

- If the field mapping is incorrect, you can drag the fields to adjust the mapping.
- If the type is automatically created at the migration destination, you need to configure the type and name of each field.

- CDM supports field conversion during the migration. For details, see **Converting Fields**.

Figure 4-122 Field mapping of Cloud Search Service

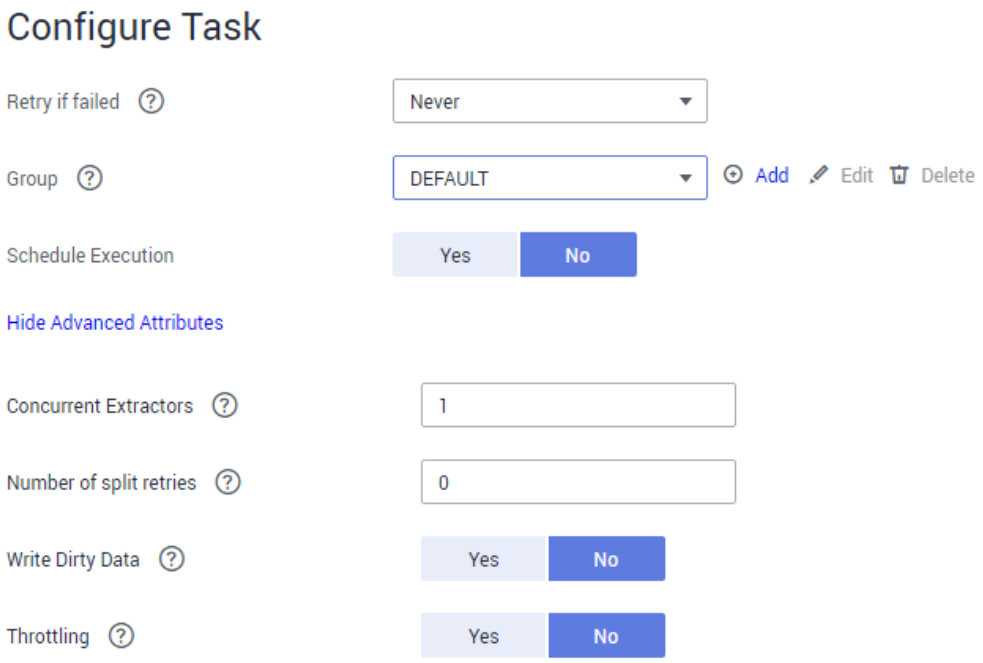


Step 3 Click **Next** and set task parameters. Generally, retain the default values of all parameters.

In this step, you can configure the following optional functions:

- **Retry If Failed**: Determine whether to automatically retry the job if it fails. Retain the default value **Never**.
- **Group**: Select the group to which the job belongs. The default group is **DEFAULT**. On the **Job Management** page, jobs can be displayed, started, or exported by group.
- **Schedule Execution**: Determine whether to automatically execute the job at a scheduled time. Retain the default value **No** in this example.
- **Concurrent Extractors**: Enter the number of concurrent extractors. An appropriate value improves migration efficiency. For details, see **Performance Tuning**. Retain the default value **1**.
- **Write Dirty Data**: Specify this parameter if data that fails to be processed or filtered out during job execution needs to be written to OBS for future viewing. Before writing dirty data, create an OBS link on the CDM console. Retain the default value **No** so that dirty data is not recorded.

Figure 4-123 Configuring the task



Step 4 Click **Save and Run**. The **Job Management** page is displayed, on which you can view the job execution progress and result.

Step 5 After the job is successfully executed, in the **Operation** column of the job, click **Historical Record** to view the job's historical execution records and read/write statistics.

On the **Historical Record** page, click **Log** to view the job logs.

----End

4.10.10 Migrating Data from OBS to DLI

Scenario

DLI is a fully hosted big data query service. This section describes how to use CDM to migrate data from OBS to DLI. The procedure includes four steps:

1. **Creating a CDM Cluster**
2. **Creating a DLI Link**
3. **Creating an OBS Link**
4. **Creating a Migration Job**

Prerequisites

- You have enabled OBS and DLI and have the permissions to read data from OBS.
- You have created resource queues, databases, and tables on DLI.

Creating a CDM Cluster

If CDM is used an independent service, create a CDM cluster by following the instructions in **Creating a CDM Cluster**. If CDM is used as a module of DataArts Studio, create a CDM cluster by following the instructions in **Creating a CDM Cluster**.

In this scenario, if the CDM cluster is used only to migrate data from OBS to DLI and does not need to migrate data of other data sources, there is no special requirements on the VPC, subnet, and security group of the CDM cluster. You can specify them based on your needs. CDM accesses DLI and OBS through the intranet. The flavor of the CDM cluster is selected based on the amount of data to be migrated. Generally, cdm.medium meets the requirements for most migration scenarios.

Creating a DLI Link

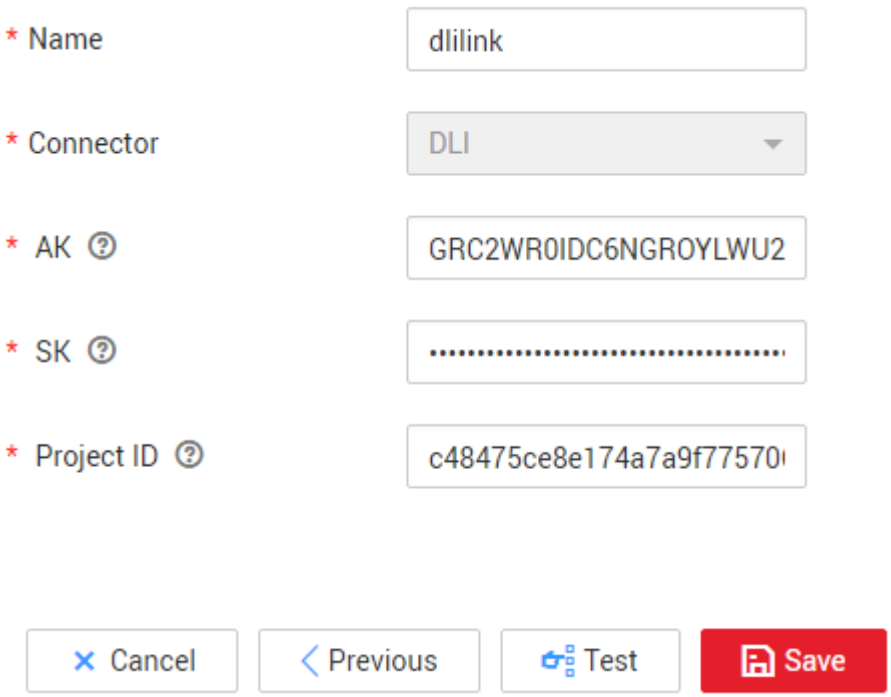
Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 4-124 Selecting a connector

Step 2 Select **Data Lake Insight**, click **Next**, and configure the DLI link parameters. See **Figure 4-125**.

- **Name**: Enter a custom link name, for example, **dlilink**.
- **AK** and **SK**: Enter the AK and SK used for accessing the DLI database.
- **Project ID**: Enter the project ID of the region to which DLI belongs.

Figure 4-125 Creating a DLI link



Step 3 Click **Save**. The **Link Management** page is displayed.

----End

Creating an OBS Link

Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 4-126 Selecting a connector type

Step 2 Select **Object Storage Service (OBS)** and click **Next** to configure parameters for the OBS link.

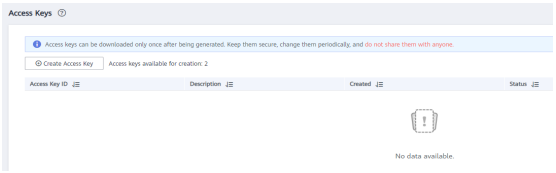
- **Name**: Enter a custom link name, for example, **obslink**.
- **OBS Server** and **Port**: Enter the actual OBS address information.
- **AK** and **SK**: Enter the AK and SK used for logging in to OBS.

  To obtain an access key, perform the following steps:

  a. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list.

  b. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 4-127**.

  Figure 4-127 Clicking Create Access Key

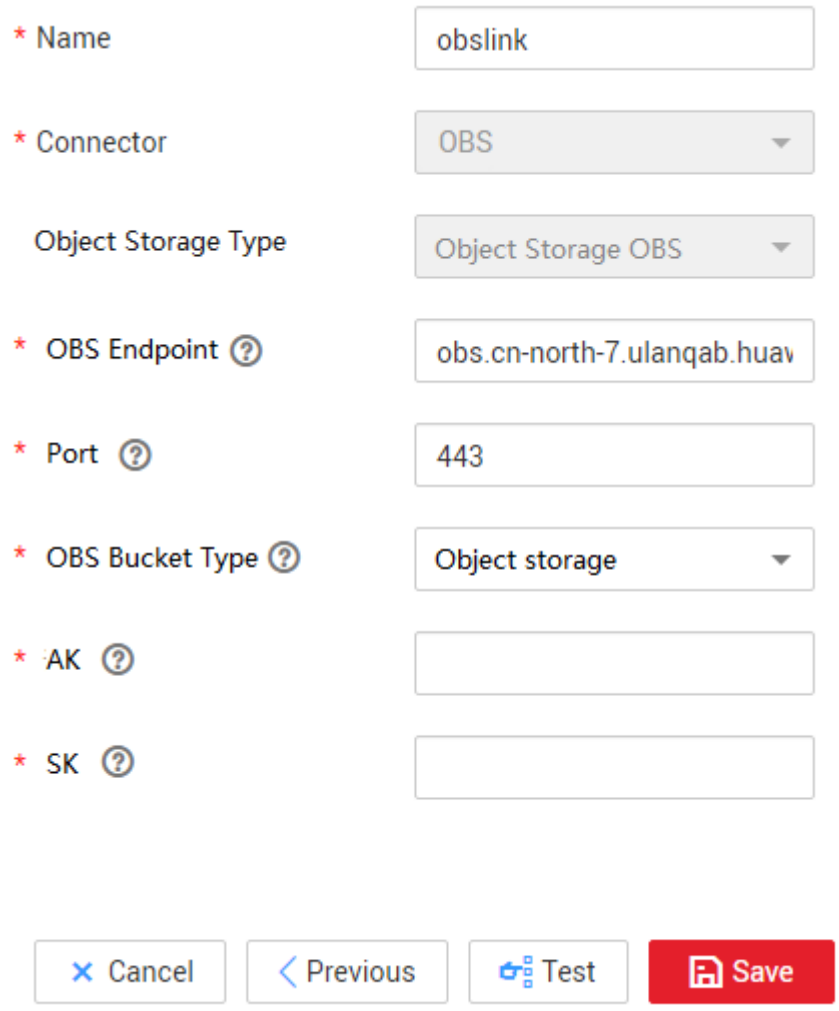
  

  c. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**.

  NOTE

  - Only two access keys can be added for each user.
  - To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.

Figure 4-128 Creating an OBS link



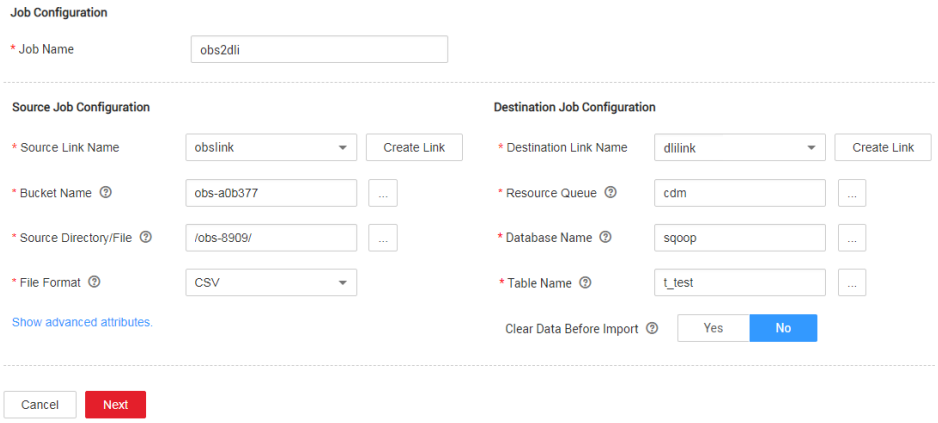
Step 3 Click **Save**. The **Link Management** page is displayed.

----End

Creating a Migration Job

Step 1 Choose **Table/File Migration** > **Create Job** to create a job for migrating data from OBS to DLI. See **Figure 4-129**.

Figure 4-129 Creating a job for migrating data from OBS to DLI



- **Job Name**: Enter a custom job name.
- **Source Link Name**: Select the **obslink** link created in **Creating an OBS Link**.
  - **Bucket Name**: Select the bucket from which the data is to be migrated.
  - **Source Directory/File**: Set this parameter to the path of the data to be migrated.
  - **File Format**: Select **CSV** or **JSON** for transferring files to a data table.
  - Retain the default values of the optional parameters in **Show Advanced Attributes**.
- **Destination Link Name**: Select the **dlilink** link created in **Creating a DLI Link**.
  - **Resource Queue**: Enter the resource queue to which the destination table belongs.
  - **Database Name**: Enter the name of the database to which data is to be written.
  - **Table Name**: Enter the name of the table to which data is to be written. CDM cannot automatically create tables on DLI. The table must be created on DLI in advance, and the field types and formats of the table must be consistent with those of the data to be migrated.
  - **Clear Before Importing Data**: Choose whether to clear data in the destination table before data import. In this example, retain the default value.

Step 2 Click **Next**. The **Map Field** page is displayed. CDM automatically matches the source and destination fields.

- If the field mapping is incorrect, you can drag the fields to adjust the mapping.
- CDM supports field conversion during the migration. For details, see **Converting Fields**.

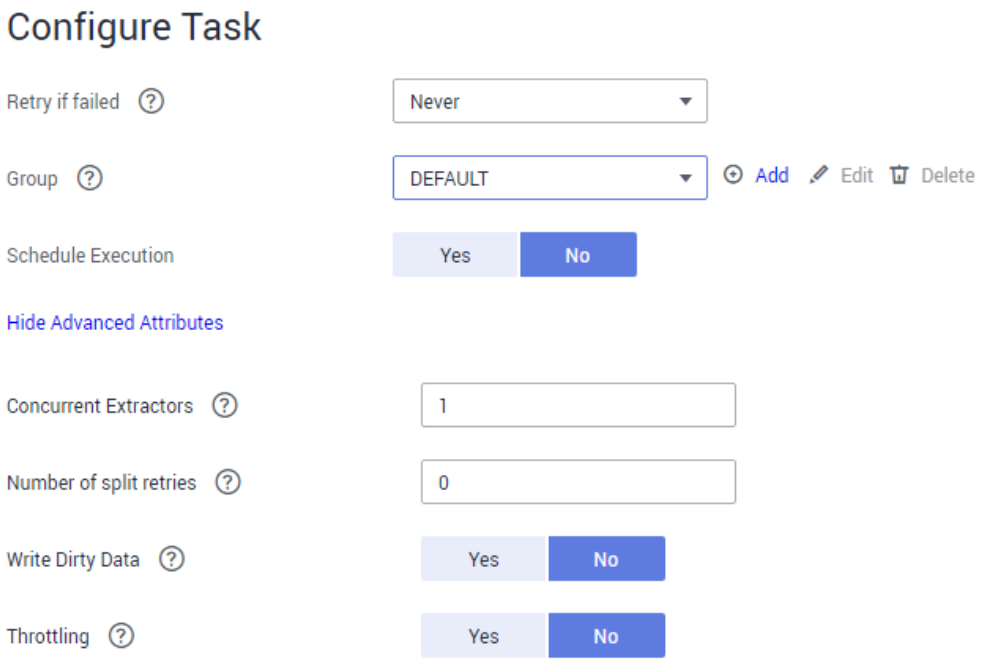
Step 3 Click **Next** and set task parameters. Generally, retain the default values of all parameters.

In this step, you can configure the following optional functions:

- **Retry If Failed**: Determine whether to automatically retry the job if it fails. Retain the default value **Never**.

- **Group**: Select the group to which the job belongs. The default group is **DEFAULT**. On the **Job Management** page, jobs can be displayed, started, or exported by group.
- **Schedule Execution**: Determine whether to automatically execute the job at a scheduled time. Retain the default value **No** in this example.
- **Concurrent Extractors**: Enter the number of concurrent extractors. An appropriate value improves migration efficiency. For details, see **Performance Tuning**. Retain the default value **1**.
- **Write Dirty Data**: Specify this parameter if data that fails to be processed or filtered out during job execution needs to be written to OBS for future viewing. Before writing dirty data, create an OBS link on the CDM console. Retain the default value **No** so that dirty data is not recorded.

Figure 4-130 Configuring the task



Step 4 Click **Save and Run**. The **Job Management** page is displayed, on which you can view the job execution progress and result.

Step 5 After the job is successfully executed, in the **Operation** column of the job, click **Historical Record** to view the job's historical execution records and read/write statistics.

On the **Historical Record** page, click **Log** to view the job logs.

----End

4.10.11 Migrating Data from MRS HDFS to OBS

Scenario

CDM supports file-to-file data migration. This section describes how to migrate data from MRS HDFS to OBS. The process is as follows:

1. **Creating a CDM Cluster and Binding an EIP to the Cluster**
2. **Creating an MRS HDFS Link**
3. **Creating an OBS Link**
4. **Creating a Migration Job**

Prerequisites

- You have obtained the domain name, port number, AK, and SK for accessing OBS.
- You have purchased an MRS cluster.
- Your EIP quota is sufficient.

Creating a CDM Cluster and Binding an EIP to the Cluster

Step 1 If CDM is used an independent service, create a CDM cluster by following the instructions in **Creating a CDM Cluster**. If CDM is used as a module of DataArts Studio, create a CDM cluster by following the instructions in **Creating a CDM Cluster**.

The key configurations are as follows:

- The flavor of the CDM cluster is selected based on the amount of data to be migrated. Generally, cdm.medium meets the requirements for most migration scenarios.
- The VPC, subnet, and security group of the CDM cluster must be the same as those of the MRS cluster.

Step 2 After the CDM cluster is created, on the **Cluster Management** page, click **Bind EIP** in the **Operation** column to bind an EIP to the cluster. The CDM cluster uses the EIP to access MRS HDFS.

NOTE

If SSL encryption is configured for the access channel of a local data source, CDM cannot connect to the data source using the EIP.

----End

Creating an MRS HDFS Link

Step 1 On the **Cluster Management** page, locate a cluster and click **Job Management** in the **Operation** column. On the displayed page, click the **Links** tab and then **Create Link**.

Figure 4-131 Selecting a connector type

Step 2 Select **MRS HDFS** and click **Next** to configure parameters for the MRS HDFS link.

- **Name**: Enter a custom link name, for example, **mrs_hdfs_link**.
- **Manager IP**: IP address of MRS Manager. Click **Select** next to the **Manager IP** text box to select a created MRS cluster. CDM automatically fills in the authentication information.
- **Username**: If **Authentication Method** is set to **KERBEROS**, set the username and password for logging in to MRS Manager.

  If you need to create a snapshot when exporting a directory from HDFS, the user configured here must have the administrator permission on HDFS.
- **Password**: password for logging in to MRS Manager
- **Authentication Method**: authentication method for accessing MRS
- **Run Mode**: Select the running mode of the HDFS link.

----End

Creating an OBS Link

Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 4-132 Selecting a connector type

Step 2 Select **Object Storage Service (OBS)** and click **Next** to configure parameters for the OBS link.

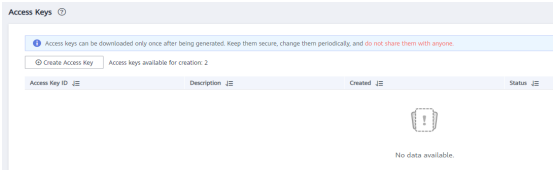
- **Name**: Enter a custom link name, for example, **obslink**.
- **OBS Server** and **Port**: Enter the actual OBS address information.
- **AK** and **SK**: Enter the AK and SK used for logging in to OBS.

  To obtain an access key, perform the following steps:

  a. Log in to the management console, move the cursor to the username in the upper right corner, and select **My Credentials** from the drop-down list.

  b. On the **My Credentials** page, choose **Access Keys**, and click **Create Access Key**. See **Figure 4-133**.

  Figure 4-133 Clicking Create Access Key

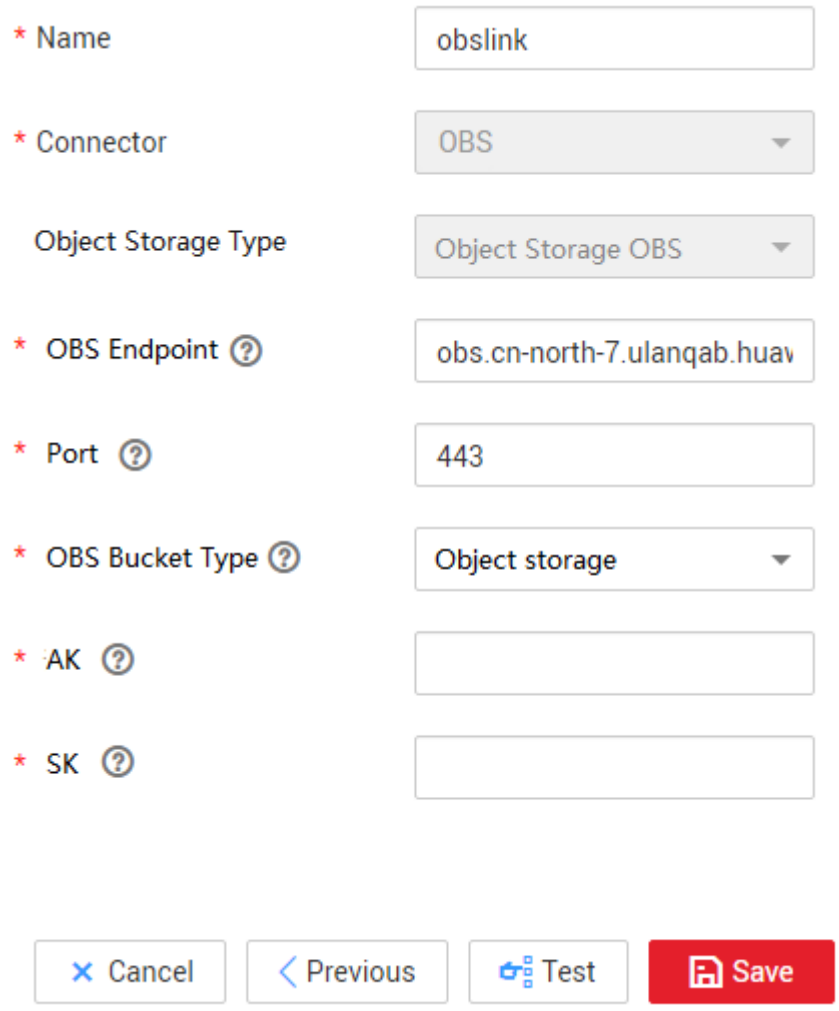
  

  c. Click **OK** and save the access key file as prompted. The access key file will be saved to your browser's configured download location. Open the **credentials.csv** file to view **Access Key Id** and **Secret Access Key**.

  NOTE

  - Only two access keys can be added for each user.
  - To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.

Figure 4-134 Creating an OBS link



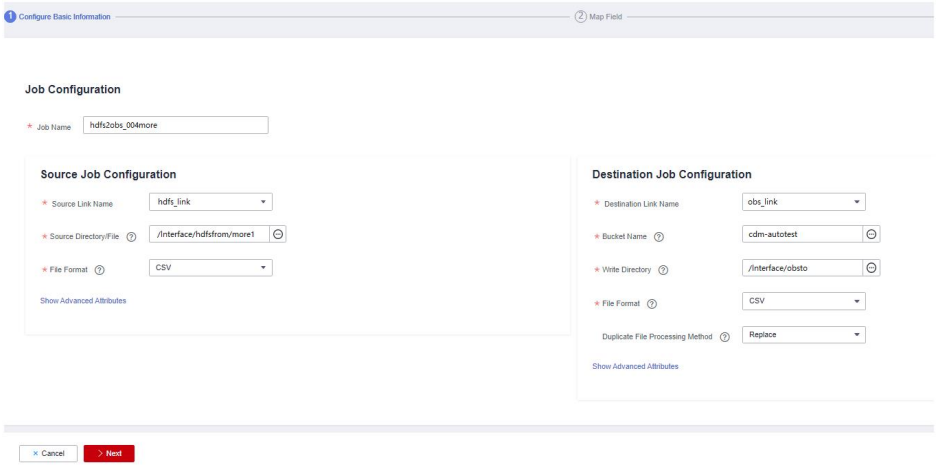
Step 3 Click **Save**. The **Link Management** page is displayed.

----End

Creating a Migration Job

Step 1 Choose **Table/File Migration** > **Create Job** to create a job for exporting data from the MRS HDFS database to OBS.

Figure 4-135 Creating a job for migrating data from MRS HDFS to OBS



- **Job Name**: Enter a unique name.
- **Source Job Configuration**
  - **Source Link Name**: Select the **hdfs_llink** created in **Creating an MRS HDFS Link**.
  - **Source Directory/File**: Enter the directory or file path of the data to be migrated.
  - **File Format**: Select the file format used for data transmission. Select **Binary**. If files are transferred without being parsed, the file format does not have to be **Binary**. This applies to file copy.
  - Retain the default values of other optional parameters.
- **Destination Job Configuration**
  - **Destination Link Name**: Select the **obs_link** created in **Creating an OBS Link**.
  - **Bucket Name**: Select the bucket from which the data will be migrated.
  - **Write Directory**: Enter the directory to which data is to be written on the OBS server.
  - **File Format**: Select **Binary**.
  - Retain the default values of the optional parameters in **Show Advanced Attributes**.

Step 2 Click **Next**. The **Map Field** page is displayed. CDM automatically matches the source and destination fields.

- If the field mapping is incorrect, you can drag the fields to adjust the mapping.
- The expressions in CDM support field conversion of common character strings, dates, and values. For details, see **Converting Fields**.

Step 3 Click **Next** and set task parameters. Generally, retain the default values of all parameters.

In this step, you can configure the following optional functions:

- **Retry Upon Failure**: If the job fails to be executed, you can determine whether to automatically retry. Retain the default value **Never**.

- **Group**: Select the group to which the job belongs. The default group is **DEFAULT**. On the **Job Management** page, jobs can be displayed, started, or exported by group.
- **Schedule Execution**: Enable it if you need to configure scheduled jobs. Retain the default value **No**.
- **Concurrent Extractors**: Enter the number of extractors to be concurrently executed. CDM supports concurrent extraction of multiple files. Increasing the value of this parameter can improve migration efficiency.
- **Write Dirty Data**: Select **No**. The file-to-file migration is binary, and no dirty data will be generated.
- **Delete Job After Completion**: Retain the default value **Do not delete**. You can also set this parameter to **Delete** to prevent an accumulation of too many migration jobs.

Step 4 Click **Save and Run**. The **Job Management** page is displayed, on which you can view the job execution progress and result.

Step 5 After the job is successfully executed, in the **Operation** column of the job, click **Historical Record** to view the job's historical execution records and read/write statistics.

On the **Historical Record** page, click **Log** to view the job logs.

----End

4.10.12 Migrating the Entire Elasticsearch Database to CSS

Scenario

CSS provides users with structured and unstructured data search, statistics, and report capabilities. This section describes how to use CDM to migrate the entire Elasticsearch database to Cloud Search Service. The procedure is as follows:

1. **Creating a CDM Cluster and Binding an EIP to the Cluster**
2. **Creating a Cloud Search Service Link**
3. **Creating an Elasticsearch Link**
4. **Creating an Entire DB Migration Job**

Prerequisites

- You have sufficient EIP quota.
- You have subscribed to CSS and obtained the IP address and port number of the CSS cluster.
- You have obtained the IP address, port number, username, and password of the on-premises Elasticsearch database server.

  If the Elasticsearch server is deployed on an on-premises data center or a third-party cloud, ensure that an IP address that can be accessed from the public network has been configured for the Elasticsearch server, or the VPN or Direct Connect between the on-premises data center and HUAWEI CLOUD has been established.

Creating a CDM Cluster and Binding an EIP to the Cluster

Step 1 If CDM is used an independent service, create a CDM cluster by following the instructions in **Creating a CDM Cluster**. If CDM is used as a module of DataArts Studio, create a CDM cluster by following the instructions in **Creating a CDM Cluster**.

The key configurations are as follows:

- The flavor of the CDM cluster is selected based on the amount of data to be migrated. Generally, cdm.medium meets the requirements for most migration scenarios.
- The CDM and Cloud Search Service clusters must be in the same VPC. In addition, it is recommended that the CDM cluster be in the same subnet and security group as the Cloud Search Service cluster.
- If the same subnet and security group cannot be used for security purposes, ensure that a security group rule has been configured to allow the CDM cluster to access the Cloud Search Service cluster.

Step 2 After the CDM cluster is created, on the **Cluster Management** page, click **Bind EIP** in the **Operation** column to bind an EIP to the cluster. The CDM cluster uses the EIP to access the on-premises Elasticsearch.

NOTE

If SSL encryption is configured for the access channel of a local data source, CDM cannot connect to the data source using the EIP.

----End

Creating a Cloud Search Service Link

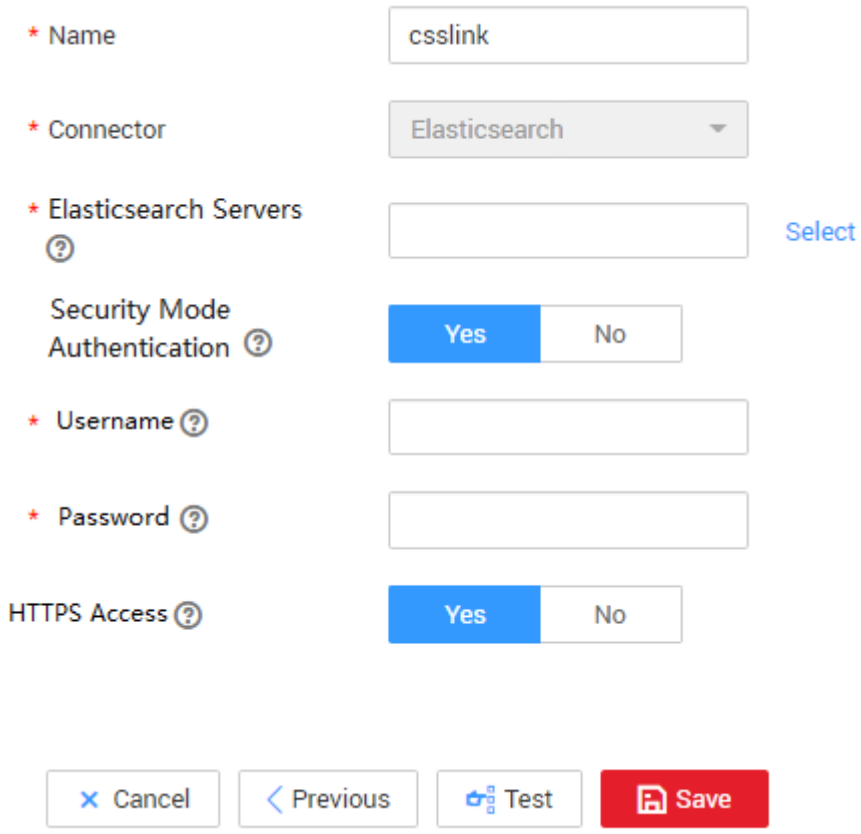
Step 1 Click **Job Management** in the **Operation** column of the CDM cluster. On the displayed page, click the **Links** tab and then **Create Link**. The **Select Connector** page is displayed.

Figure 4-136 Selecting a connector

Step 2 Select **Cloud Search Service** and click **Next**. On the page that is displayed, configure the CSS link parameters.

- **Name**: Enter a custom link name, for example, **csslink**.
- **Elasticsearch Server List**: Enter the IP address and port number of the Cloud Search Service cluster (cluster later than 5.*x*). The format is *ip:port*. Use semicolons to separate multiple addresses. For example, **192.168.0.1:9200;192.168.0.2:9200**.
- **Username** and **Password**: Enter the username and password used for logging in to the Cloud Search Service cluster. The user must have the read and write permissions on the database.

Figure 4-137 Creating a CSS link



Step 3 Click **Save**. The **Link Management** page is displayed.

----End

Creating an Elasticsearch Link

Step 1 On the **Cluster Management** page, locate a cluster and click **Job Management** in the **Operation** column. On the displayed page, click the **Links** tab and then **Create Link**.

Figure 4-138 Selecting a connector type

Step 2 Select **Elasticsearch** and click **Next** to configure parameters for the Elasticsearch link. The parameters are the same as those for the CSS link.

- **Name**: Enter a custom link name, for example, **es_link**.
- **Elasticsearch Server List**: Enter the IP address and port number of the on-premises Elasticsearch database. Use semicolons to separate multiple addresses.

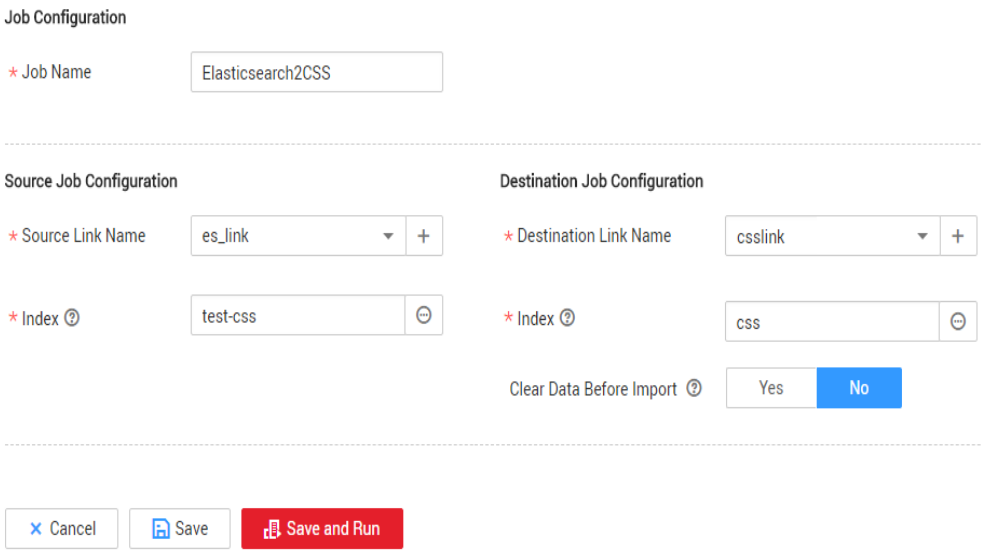
Step 3 Click **Save**. The **Link Management** page is displayed.

----End

Creating an Entire DB Migration Job

Step 1 Choose **Entire DB Migration** > **Create Job** to create an entire DB migration job.

Figure 4-139 Creating an entire DB migration job



- **Job Name**: Enter a unique name.

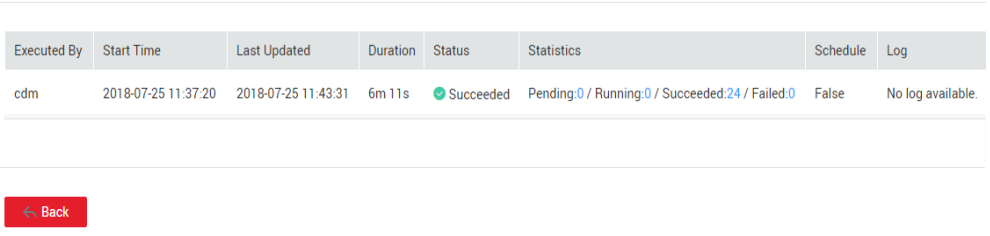
- **Source Job Configuration**
  - **Source Link Name**: Select the **es_link** link created in **Creating an Elasticsearch Link**.
  - **Index**: Click the icon next to the text box to select an index in the on-premises Elasticsearch database or manually enter an index name. The name can contain only lowercase letters. If multiple indexes need to be migrated at a time, set this parameter to a wildcard character. CDM migrates all indexes that meet the wildcard condition. For example, if this parameter is set to **cdm***, CDM migrates all indexes starting with **cdm**, such as **cdm01**, **cdmB3**, **cdm_45** and so on.
- **Destination Job Configuration**
  - **Destination Link Name**: Select the **csslink** link created in **Creating a Cloud Search Service Link**.
  - **Index**: Enter the index of the data to be written. You can select an existing index in Cloud Search Service or manually enter an index name that does not exist. The name can contain only lowercase letters. CDM automatically creates the index in Cloud Search Service. If multiple indexes are migrated at a time, this parameter cannot be configured. CDM automatically creates indexes at the migration destination.
  - **Clear Data Before Import**: If the selected index already exists in Cloud Search Service, you can choose whether to clear the data in the index before importing data. If you select **No**, the data is added to the index.

Step 2 Click **Save and Run**. The **Job Management** page is displayed, on which you can view the job execution progress and result.

A sub-job will be generated for each type in the on-premises Elasticsearch index for concurrent execution. You can click the job name to view the sub-job progress.

Step 3 After the job is successfully executed, in the **Operation** column of the job, click **Historical Record** to view the job's historical execution records, read/write statistics, and job logs (only the sub-jobs have job logs).

Figure 4-140 Historical Record



----End

4.10.13 More Cases and Practices

For more advanced guidance and cases of DataArts Migration, see **Best Practices**.

5 DataArts Architecture

5.1 Overview

Model Design Method Overview

A data model can reflect the relationships between objects. It incorporates the key information features extracted based on business requirements. It visually represents how the internal information of an enterprise is organized. A data model must be capable of simulating scenarios, easy-to-understand, and easily implemented in the IT system.

ER and dimensional modeling are both used on DataArts Architecture.

- **ER modeling**

  ER modeling describes the business processes within an enterprise. Compliant with the third normal form (3NF), ER modeling is designed for data integration. It is used for combining and merging data with similarities by subject. ER modeling results cannot be used directly for decision-making, but they are a useful tool.

  There are three different models involved in ER modeling: design conceptual models, logical models, and physical models.

  - **Conceptual model** is used to represent business processes and business data involved in various activities. A conceptual model illustrates the relationships between business entities.
  - **Logical model** is much more detailed than the conceptual model. Logical models outline business details based on entities, attributes, and relationships. They enable communication between IT and business staff. A logical model is a set of standardized logical table structures. Based on business rules, a logical model outlines business objects, data items of the business objects, and relationships between business objects.
  - **Physical model**: An advanced version of the logic model and used to design the database architecture for data storage with a full consideration of various technical factors. For example, the selected data warehouse is DWS or MRS_Hive.

  - The system distinguishes physical models from logical models. For example, after you access a physical model, the drop-down list on the left contains only physical models.

- **Dimensional modeling**

  Dimensional modeling is the construction of models based on analysis and decision-making requirements. It is mainly used for data analysis. Dimensional modeling is focused on how to quickly analyze user requirements and respond rapidly to complicated, large-scale queries.

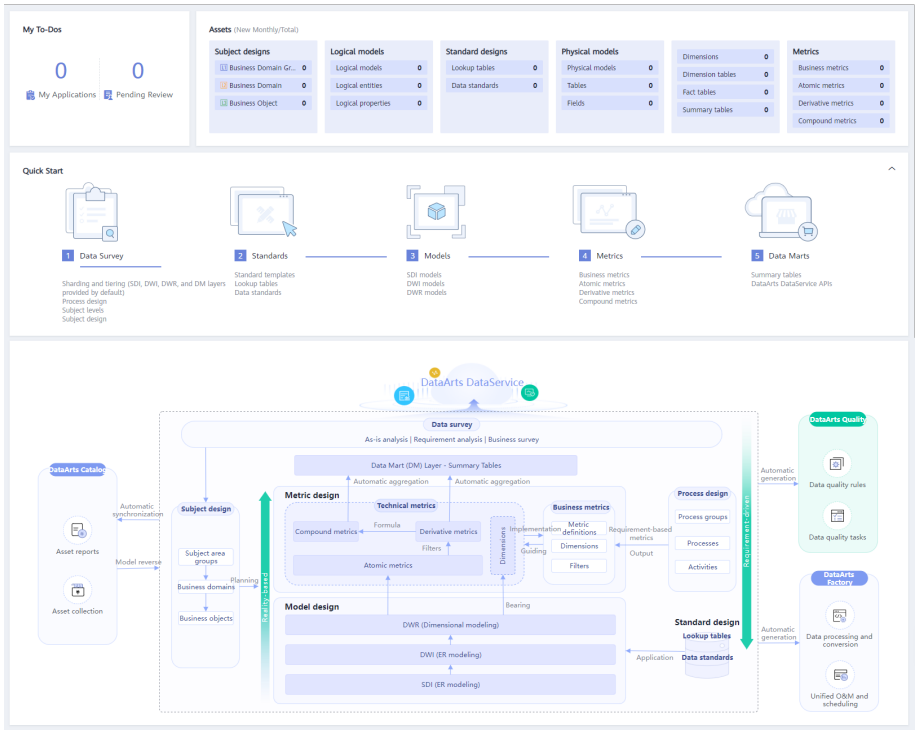
  A multidimensional model is a fact table consisting of numeric metrics. The fact table is associated with a group of dimensional tables containing description attributes with primary or foreign keys. Typical dimensional models include star models and snowflake models used in some special scenarios.

  In the DataArts Architecture module of DataArts Studio, dimensional modeling involves abstracting facts and dimensions for model creation, and abstracting and sorting out report requirements for constructing metric systems and creating summary models.

DataArts Architecture Overview Page

On the DataArts Studio console, locate a workspace and click **DataArts Architecture**. The **Overview** page is displayed.

Figure 5-1 DataArts Architecture Overview page



- **My To-Dos**
  - The **My To-Dos** area displays the quantity of **My Applications** and **Pending Review**.

  - Click the numbers above **My Applications** and **Pending Review** to access the **My Applications** and **Pending Review** pages, respectively.
- **Assets**
  - The **Assets** area displays all the objects in DataArts Architecture.
  - Click the number next to each object name to access the object management page.
- **Quick Start**

  The **Quick Start** area displays the overall process for data governance. You can click a specific operation under the process to go to the corresponding page.
- **DataArts Architecture Process**
  - This area displays the DataArts Architecture process and how the DataArts Architecture module interacts with other modules of DataArts Studio. For details about the DataArts Architecture process, see **DataArts Architecture Use Process**.
  - You can move the cursor over the name of an object to view its description.
  - You can click the name of any object supported by DataArts Studio to access the object management page.

Information Architecture of DataArts Architecture

An information architecture is a set of component specifications that describe various types of information required for business operations and management decision-making as well as the relationships of business entities. On the **Information Architecture** page, you can view and manage all tables, including business tables, dimension tables, fact tables, and summary tables.

On the DataArts Studio console, locate a workspace and click **DataArts Architecture**. In the navigation pane, choose **Information Architecture**.

Perform the following operations on the **Information Architecture** page.

- **Search**

  On the top of the **Information Architecture** page, click **Advanced Search**, set the table name, type, data source, and other filters, and click **Search** to search for a specific table. Then click the table name to access its details page.
- **Create**

  Click **Create** to create a logical model, physical model, dimension table, fact table, or summary table. For details, see **Designing Logical Models**, **Designing Physical Models**, **Creating Dimensions**, **Creating Fact Tables**, or **Creating Summary Tables**.
- **Synchronize**

  Choose **More** > **Synchronize** to synchronize tables to DataArts Catalog as technical assets or synchronize logical models to DataArts Catalog as logical assets. You can choose to synchronize them to the production or development environment. By default, they are synchronized to the production environment.
- **Modify Subject**

Choose **More** > **Modify Subject** to change the selected table to another subject.

- **Delete**

  Choose **More** > **Delete** to delete a data table. A data table in publishing review, published, or suspension review state cannot be deleted. A referenced data table cannot be deleted either.

- **Suspend**

  Choose **More** > **Suspend** to suspend a published data table. A referenced data table cannot be suspended.

  NOTE

  Edited versions refer to the data that is re-edited after the publishing review.

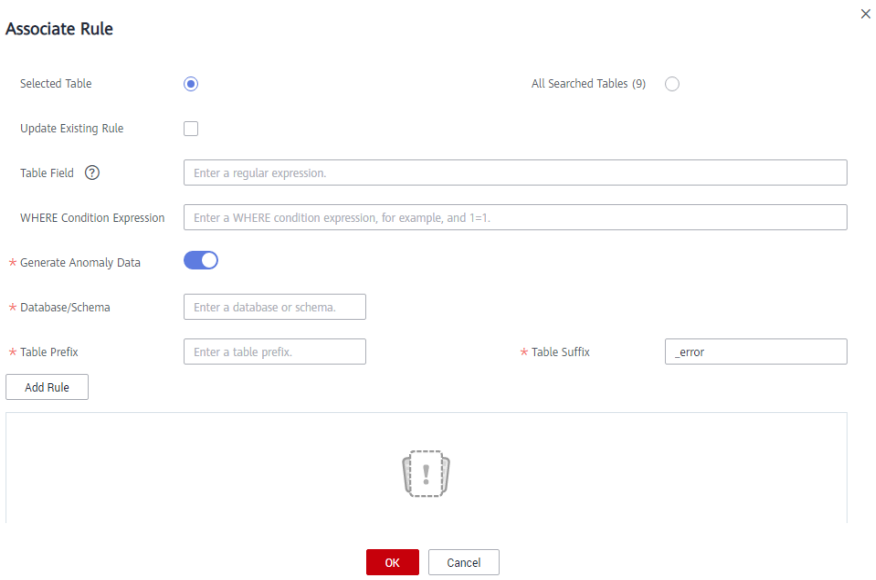
- **Publish**

  Click **Publish** to publish a data table. Data tables in publishing review, suspension review, or published (without edited versions) state cannot be published. You can choose to publish data tables to the production or development environment. By default, they are synchronized to the production environment.

- **Associate Rule**

  Click **Associate Rule** and set the parameters to associate a quality rule with the object you select. For details, see **Associating Quality Rules**.

  Figure 5-2 Associating a quality rule with an object

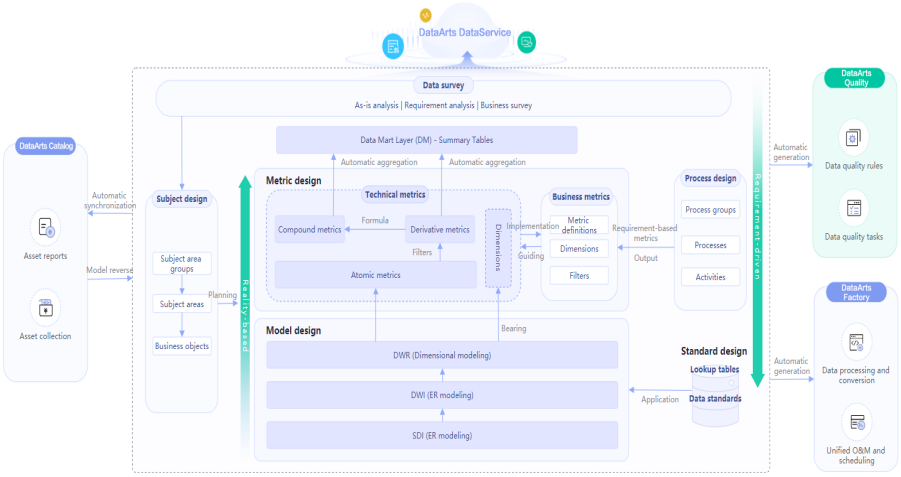


  Generate Anomaly Data: If this option is selected, anomaly data is stored in the specified database based on the configured parameters.

5.2 DataArts Architecture Use Process

The process of using DataArts Architecture is as follows.

Figure 5-3 DataArts Architecture use process



1. **Preparations**
   - **Add reviewers**: In the DataArts Architecture module, all business processes must be approved. Therefore, add reviewers first before conducting any operations. Only the workspace admin has the permissions required to add reviewers.
   - **Configuration Center** provides abundant custom options. You can customize the configuration to meet your demands.
2. **Data Survey**: A data survey involves collecting data that is generated when sorting business requirements, creating business processes, and classifying data subjects based on the existing business data and industry status.
   - **Subject design** is a hierarchical architecture that classifies and defines data to help clarify data assets and specify relationships between business domains and business objects.
     - **Subject area group** is used to group business domains based on scenarios.
     - **Subject area** is the high-level data classification that does not overlap and is used to manage business objects.
     - **Business object** includes important information about people, events, and things that are indispensable to enterprise operations and management.
   - **Process design** is used to generate a structured framework of process. It describes the categories, levels, boundaries, scopes, and input/output relationships of an enterprise's processes, and reflects the business models and characteristics of the enterprise.
3. **Standards**: Create lookup tables and data standards.
   - A **lookup table** includes a series of allowed values and additional text descriptions that are generally associated with data standards to generate a range of values for the verification of quality monitoring rules.
   - **Data standards** refer to the description of attribute data meanings and business rules that enterprises must comply with. It describes the common understanding of certain data at the company level.

4. **Models**: Use ER modeling and dimensional modeling methods to perform hierarchical modeling.
   - **ER modeling**: Create SDI and DWI models based on ER modeling.
     - **SDI** stands for Source Data Integration and is the source data layer. SDI is a simple implementation of source system data.
     - **DWI** stands for Data Warehouse Integration, also called the data consolidation layer. DWI integrates and cleans data from multiple source systems, and implements entity relationship modeling based on the three normal forms.
   - **Dimensional modeling**: Create DWR models and release dimensions and fact tables based on ER modeling.
     - **Data Warehouse Report (DWR)** is based on the multi-dimensional model and its data granularity is the same as that of the DWI layer.
     - **Dimension** is the perspective to observe and analyze business data and assist in data aggregation, drilling, slicing, and analysis, and used as a GROUP BY condition in SQL statements.
     - A **fact table** that belongs to a business process can enrich the affair information corresponding to the specific business process.
5. **Metrics**: Create business and technical metrics. Technical metrics include atomic, derivative, and compound metrics.
   - A **metric** consists of its name and value. The metric name and its definition reflect the quality and quantity of the metric. The metric value reflects the quantifiable values of the specified time, location, and condition of the metric.

     Business metrics are used to guide technical metrics, and technical metrics are used to implement business metrics.
   - **Atomic metrics** are generated based on dimension tables and fact tables of a multidimensional model. The objects and the finest data granularity of an atomic metric are consistent with those of the multidimensional model.

     An atomic metric usually consists of measures and attributes related with measures and business objects, all of which aim to support agile self-service consumption of the metric.
   - **Derivative metrics** are aggregated from the definitions, modifiers, and dimensions of atomic metrics. Therefore, their definitions, modifiers, and dimensions are derived from the attributes of atomic metric associated tables as well.
   - **Compound metrics** are generated by adding one or more derivative metrics. The dimensions and modifiers of a compound metric are the same as those of the derivative metrics.

     New dimensions and modifiers cannot be generated outside the scope of derivative metrics, dimensions, and modifiers.
6. **Data mart**: Create a DM layer and release summary tables.
   - **Data Mart (DM)** is where multiple types of data are summarized. DM is designed to display the summarized data.

- A **summary table** consists of specific analysis objects (for example, members) and related statistical metrics. The statistical metrics included in a summary table have the same statistical granularity (for example, members). The summary table provides users with all statistics-granularity-themed data (such as a member theme market).

5.3 Preparations

5.3.1 Adding Reviewers

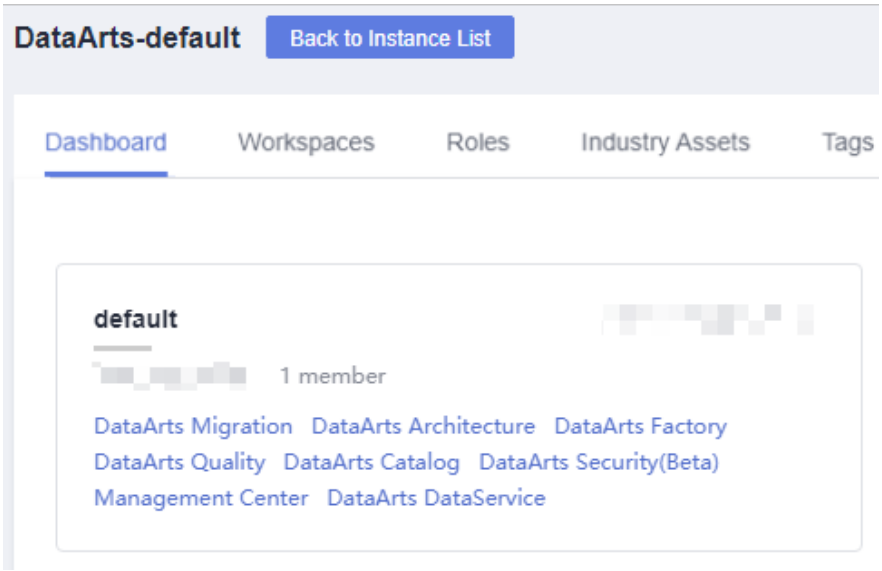
In the DataArts Architecture module, all business processes must be approved. Therefore, add reviewers first before conducting any operations. Only the workspace admin has the permissions required to add reviewers.

Adding a Reviewer

A reviewer must be a member who has the review permissions in the current workspace. You can edit and add workspace members in **Workspaces** on the DataArts Studio homepage.

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

   Figure 5-4 DataArts Architecture

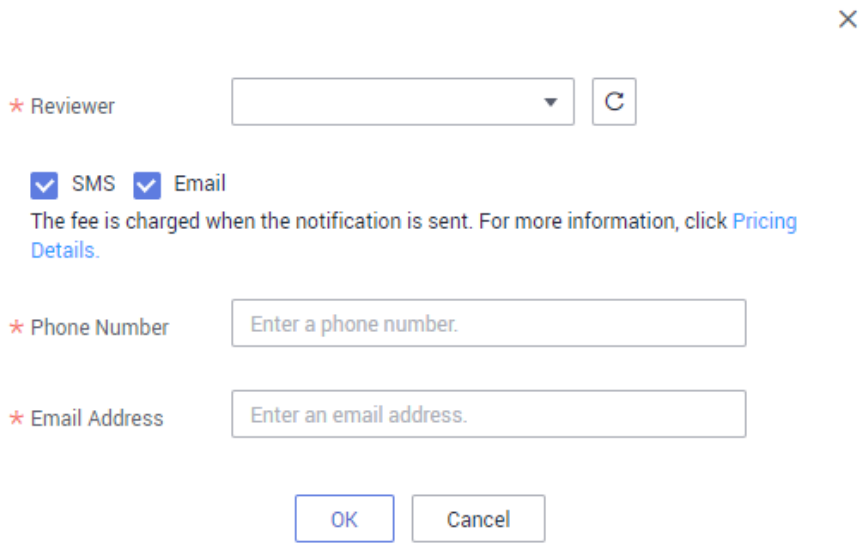
2. In the navigation pane, choose **Configuration Center**. On the displayed page, click **Reviewers**.
3. On the **Reviewer Management** tab page, click **Add**.
4. Select a reviewer, enter their mobile number and email address, and click **OK**.

   The reviewer must be admins and developers of the current workspace, because only admins and developers have the review permissions of the workspace.

NOTE

- You can only select reviewers from the given list. To enable a user to be available in the given list, add the user as a workspace member in **Workspaces** on the DataArts Studio homepage.
- If you select **SMS** or **Email** for **Notification Type**, DataArts Studio automatically creates a topic in SMN after the reviewer is added.
  - The topic name is in the following format: DataArts_Subject_Reviewer_***Project name_Project ID***-dlg_ds_***Reviewer name***.

Figure 5-5 Adding a reviewer

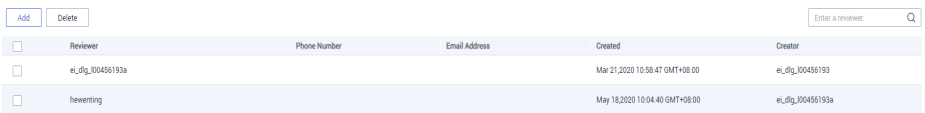


5. You can add multiple reviewers if needed.

Related Operations

On the DataArts Architecture page, choose **Configuration Center** in the left navigation pane. On the displayed page, click the **Reviewers** tab to manage reviewers.

Figure 5-6 Reviewer Management page



- **Searching for a reviewer**

  In the upper right corner of the reviewer list, enter the name of the reviewer you are looking for and click .
- **Deleting a reviewer**

  In the reviewer list, select the reviewer you want to delete, and click **Delete**.

5.3.2 Managing the Configuration Center

Constraints

The quotas for different custom objects are as follows:

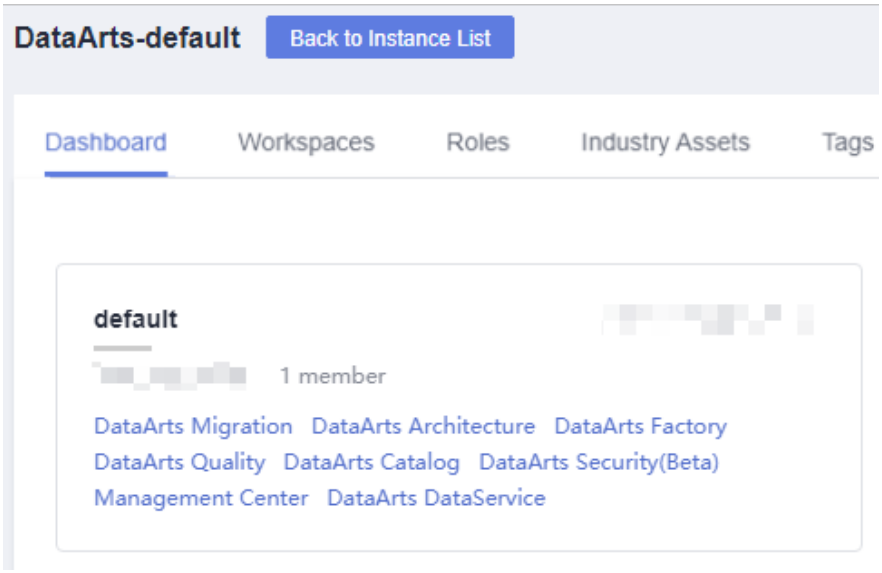
- Custom subjects: 10
- Custom tables: 30
- Custom attributes: 10
- Custom business metrics: 50

Subject Processes

You can customize the subject levels and attributes in the subject design. By default, there are three levels in the system, which are named Subject Area Group (L1), Subject Area (L2), and Business Object (L3) from top to bottom. You can define a maximum of seven levels and a minimum of two levels. You can configure a maximum of 10 custom attributes.

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

   Figure 5-7 DataArts Architecture

2. In the navigation pane, choose **Configuration Center**. On the displayed page, click the **Subject Processes** tab.
3. In the **Subject Level** area, you can add, delete, and edit subject levels.
   - Click + in the **Operation** column to add a custom subject level and click **Update**.
   - Click in the **Operation** column to delete a subject level and click **Update**.
   - Except the business object at the last level, you can click the names of other levels to edit them.

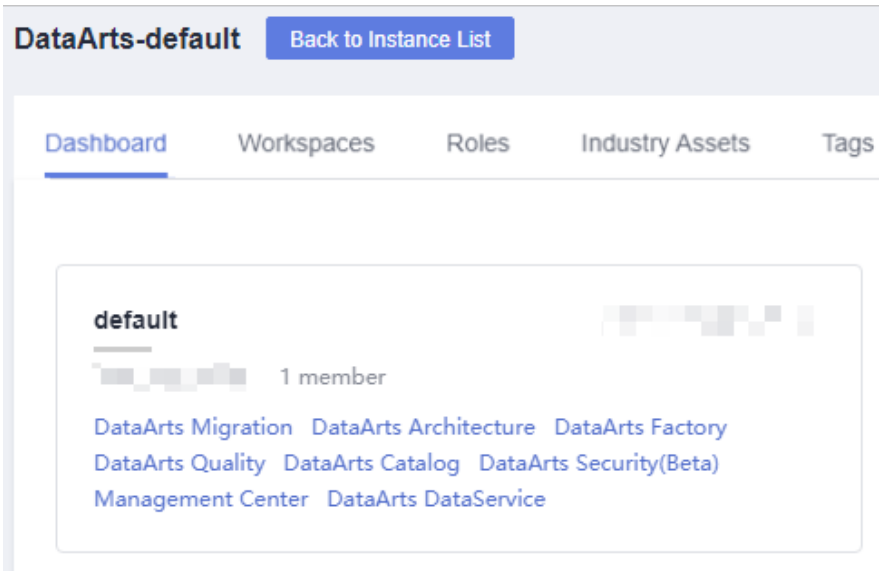
4. In the **Custom Field** area, you can create, delete, and edit fields.
   - Click **Create** to create a custom attribute.
   - Click in the **Operation** column to delete a custom attribute.
   - Edit **Field (Local)**, **Field (Eng)**, **Optional Value**, and **Mandatory**.
5. Set **Process Levels**. Enter a value from 3 to 7.

Standard Templates

You can customize the default options of data standards. When you access the **Standard Templates** page for the first time, the page for creating a data standard template is also displayed.

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

   Figure 5-8 DataArts Architecture

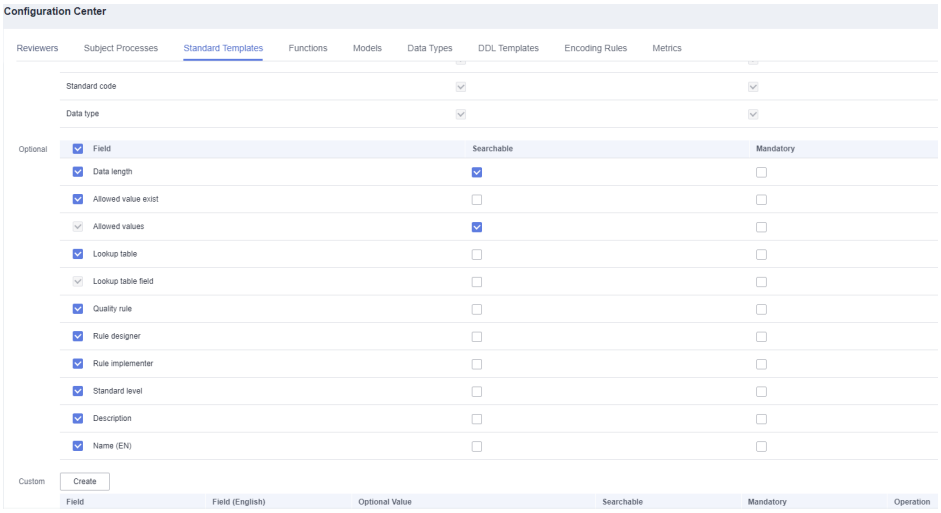


2. In the navigation pane, choose **Configuration Center**. On the displayed page, click the **Standard Templates** tab.
3. In the **Optional** area, select the parameters as required. Click **Create** next to **Custom** to add custom properties. After the configuration is complete, click **Update**.

   NOTE

   - A standard template contains the following custom fields: **Searchable**, **Mandatory**, and **Optional Value**.
   - After the template is saved, you must set values for the options selected in the template when creating a data standard.
   - When you access the **Standard Templates** page for the first time, **Data length** and **Description** are selected by default. You can select other options as needed.
   - When adding a customized item, you can add both Chinese and English items.

Figure 5-9 Standard Templates tab page

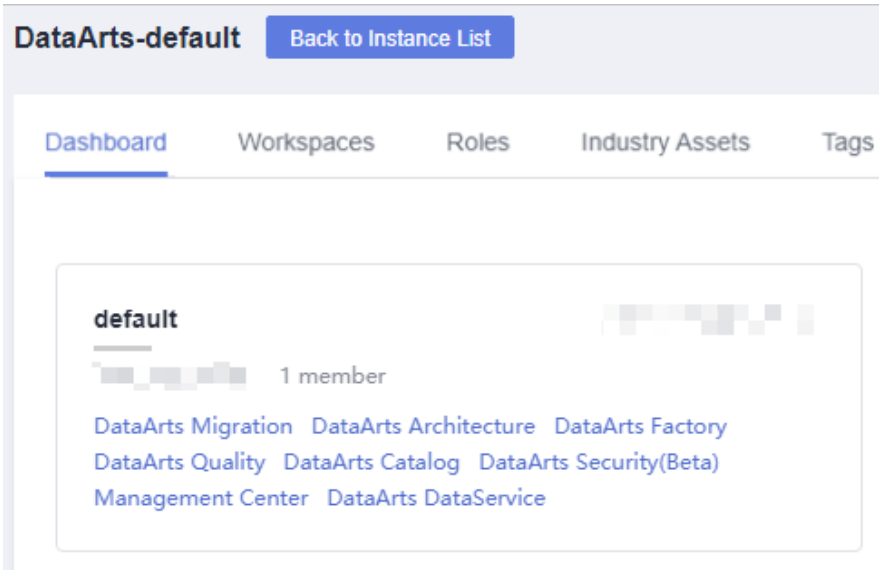


Functions

You customize functions for DataArts Architecture.

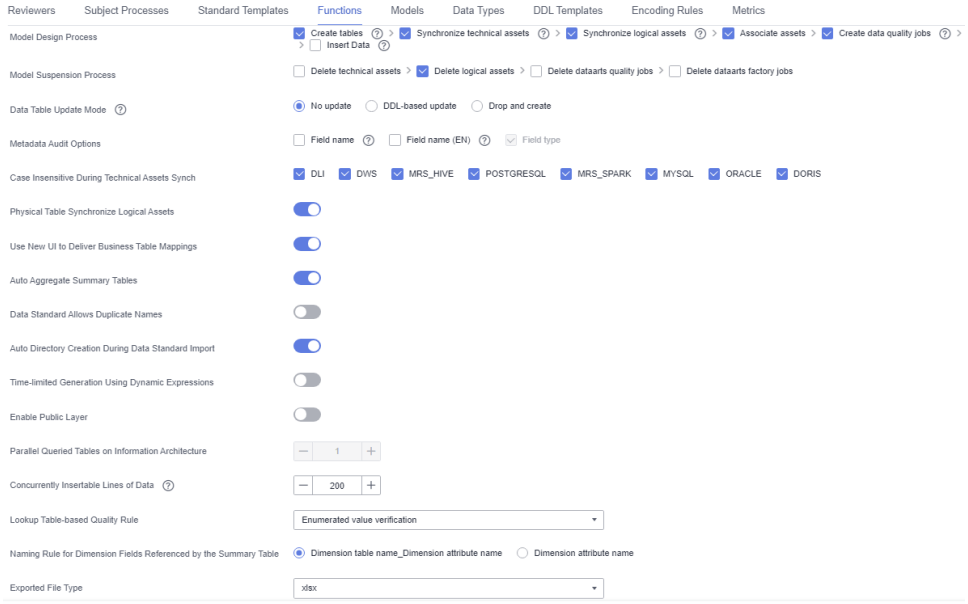
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

   Figure 5-10 DataArts Architecture

2. In the navigation pane, choose **Configuration Center**. On the displayed page, click the **Functions** tab.
3. On the page displayed, set the parameters and click **OK**. Click **Reset** to restore the default settings.

Figure 5-11 Functions



- **Model Design Process**: The selected processes are automatically executed progressively when a table created in an ER or dimension model is published and suspended. You are advised to select all the options.
  - **Create tables**: After a table publishing application is approved in DataArts Architecture, the system creates a physical table in the corresponding data source. When a table is deleted, the system deletes the corresponding physical table.
  - **Synchronize technical assets**: After a table in **ER Modeling** or **Dimensional Modeling** is published, the table is synchronized to the DataArts Catalog module as a technical asset, and the tag is synchronized to the corresponding technical asset.

    NOTE

    To enable **Synchronize Technical Assets**, you must create a data asset collection task for the database to which the table belongs in DataArts Catalog. Otherwise, the technical asset synchronization will fail.
  - **Synchronize logical assets**: The system synchronizes logical models to DataArts Catalog as logical assets. After that, the system tags the logical assets accordingly.
  - **Associate assets**: Associate logical assets with technical assets. After the logical assets and technical assets are synchronized, you can view the associated technical or logical asset when viewing the details of a logical or technical asset on the DataArts Catalog page. This function requires that the table information contains the data source information.
  - **Create data quality jobs**: After a table in **ER Modeling** or **Dimensional Modeling** is published and approved, the system automatically creates a quality job in the DataArts Quality module of DataArts Studio for a table that is associated with a data standard

(including the data length or allowed value) or associated with a quality rule.

- **Create data development jobs**: After a summary table is published, the system generates an E2E data development job.
- **Publish DataArts DataService APIs**: After a summary table is published, a DataArts DataService API is automatically generated. This function takes effect only DataArts DataService supports data connections of the summary table.
- **Insert data**: After a lookup table is published, values in the table are automatically written to the dimensional table.

– **Model Suspension Process**: Select whether to delete technical assets, logical assets, data quality jobs, and data development jobs when suspending the job.
– **Data Table Update Mode**: If a table in DataArts Architecture is modified after being published, you can choose whether to update the table in the database and how to update the table. By default, the table is not updated. However, you can set the update operation in the configuration center as required. To be more specific, configure the corresponding update statements in the DDL templates.

- **No update**: The system does not update tables in a database.
- **DDL-based update**: The system updates tables in the database based on the DDL update template configured in **DDL Templates**. The underlying data warehouse engine determines whether the update is successful. Different types of data warehouses support different table update modes. If the data warehouse does not support table update operations on the DataArts Architecture page, the tables in the database may be inconsistent with those in DataArts Architecture. For example, table fields cannot be deleted when DLI tables are updated. If table fields are deleted from the tables in DataArts Architecture, the corresponding table fields cannot be deleted from the database.

  If the offline database supports the syntax for updating the table architecture, you can configure the syntax in the DDL template. Then, the update operation can be performed. Otherwise, update the table by rebuilding it.
- **Drop and create**: The system deletes an existing table in a database and then creates a table. This option ensures that the tables in the database are the same as those in DataArts Architecture. However, since the table is deleted first, you are advised to select this option only in the development and design phase or test phase. After the product is brought online, you are not advised to select this option.

– **Case Insensitive During Technical Assets Synch**: When a table, whose type is the same as the data connection, is published, the data connection name is case insensitive during technology asset synchronization. If the name is the same as an existing one, the connection exists.
– **Physical Table Synchronize Logical Assets**: If **Synchronize logical assets** is selected and no logical asset is available, you can disable this

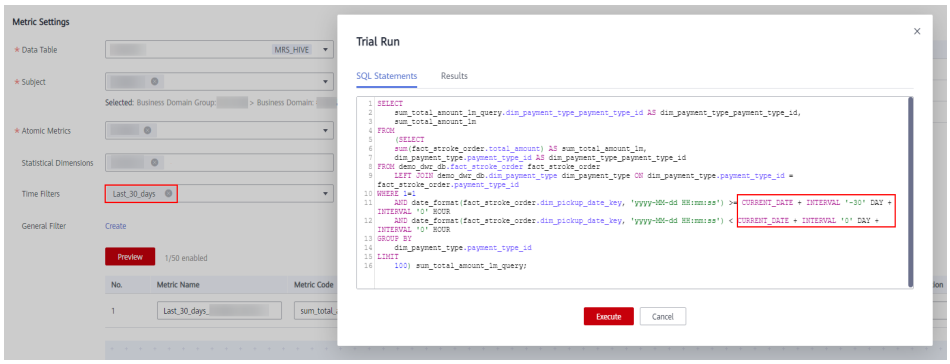
option to prevent physical tables from overwriting logical tables which have the same names as them. In this case, physical tables will only be associated with, but not synchronized to logical assets. If no logical asset is found, the association fails and an error is reported.

- **Use New UI to Deliver Business Table Mappings**: This function is enabled by default. The mapping function of the new version supports operations such as join. You are advised to use the mapping function of the new version.
- **Auto Aggregate Summary Tables**: When publishing a derivative or compound metric, the system automatically generates a summary table. A statistical dimension corresponds to a summary table. You can click the **Automatic Aggregation** tab on the summary table page to view the automatically generated summary tables.
- **Data Standard Allows Duplicate Names**: This function is disabled by default. If it is enabled, duplicate data standard names are allowed.
- **Auto Directory Creation During Data Standard Import**: This function is enabled by default.
- **Enable Public Layer**: If this option is enabled, the current workspace can be converted into a public workspace. The lookup tables and data standards of the public workspace are shared with all common workspaces. In a common workspace, you can query or reference the lookup tables and data standards of the public workspace, but cannot add, modify, or delete them.

  NOTE

  - After the current workspace is converted to a public workspace, it cannot be rolled back to a common workspace, and no other common workspaces can be converted to a public workspace. Exercise caution when selecting your public workspace.
  - You cannot query, reference, or operate the data of a common workspace from a public workspace.
- **Time-limited Generation Using Dynamic Expressions**: If you enable this function, dynamic time expressions will be used; otherwise, the default static time expressions will be used. The dynamic expression automatically updates the generated time, while the static expression does not. For example, if the current month is September and a static expression is used, data generated for the last 30 days is the data in August. Even when the current month changes to October, data generated for the last 30 days is still the data in August. However, if a dynamic expression is used, data generated for the last 30 days will automatically change to the data in September if the current month has changed to October. The following figure shows an example time function using a dynamic expression.

Figure 5-12 Dynamic expression



NOTE

If you enable this function for the first time, you need to reset the derivative metrics in the DDL template. If you have made any change to the DDL template, back up the template before resetting it. Resetting the template will overwrite any change that has been made. After the template is reset, you must make the changes again.

- **Parallel Queried Tables on Information Architecture**: The default value is **1**. Currently, you cannot change the value.
- **Concurrently Insertable Lines of Data**: The value determines the number of lines of the dimensional table into which data of the lookup table is inserted. If the lookup table contains a large amount of data, the data may fail to be inserted into the dimensional table. In this case, you can reduce the value of this parameter.
- **Lookup Table-based Quality Rule**: Select a value from the drop-down list box. If the data volume of the lookup table is small, select **Enumerated value verification**; otherwise, select **Field value consistency verification**.

  NOTE

  You can select **Field value consistency verification** only if the database contains a lookup table. The following lookup tables are contained in the database:

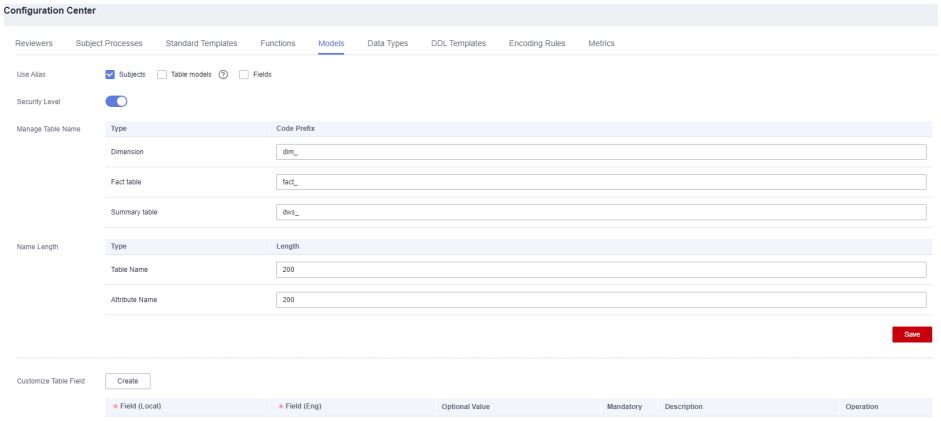
  - Lookup tables obtained by database reversion
  - Lookup tables published during dimension creation
- **Naming Rule for Dimension Fields Referenced by the Summary Table**: Set the naming rule for a summary table during creation, editing, import, and generation. Select **Dimension table name_Dimension attribute name** or **Dimension attribute name**.
- **Exported File Type**: Two options are available: **xlsx** and **et**. Logical models, physical models, dimensions (dimension tables), fact tables, summary tables, and other data can be exported in both formats.
- **Generate DataArts DataService APIs**: Two options are available: **Table API** and **Metric APIs**.

Model Settings

You can perform the following operations during subject design and model design on the **Model Settings** page.

- Add the subject alias, table model alias, and field alias.
- Set the default table code prefix for dimension tables, fact tables, and summary tables.
- Add custom fields to a table.
- Add custom fields to an attribute.

Figure 5-13 Model Settings tab page



In the navigation pane, choose **Configuration Center**. On the displayed page, click the **Models** tab.

- **Use Alias**: You can enable or disable alias.
  - The options are as follows:
    - If you select **Subjects**, you must enter an alias when creating or editing subject.
    - If you select **Table models**, you must enter an alias when creating or editing a table. Business tables, dimension tables, fact tables, and summary tables are affected when **Table models** is selected.
    - If you select **Fields**, you must enter an alias when creating or editing a table field.
- **Security Level**: Enable it. It is enabled by default.
- **Manage Table Names**: Set the default table code prefix for dimension tables, fact tables, and summary tables.
- **Customize Table Property**: When creating or editing a table, you can set custom fields in the basic settings of the table. Business tables, dimension tables, fact tables, and summary tables are affected.
- **Customize Attribute Field**: When creating or editing a table field, you can set custom attributes in the table field. Business tables, dimension tables, fact tables, and summary tables are affected.

Field Types

When you create a table, reverse a database, or convert a model, if the default data type or the data type mappings between different data sources cannot meet

your requirements, you can add, delete, or modify data types. The default data type cannot be deleted.

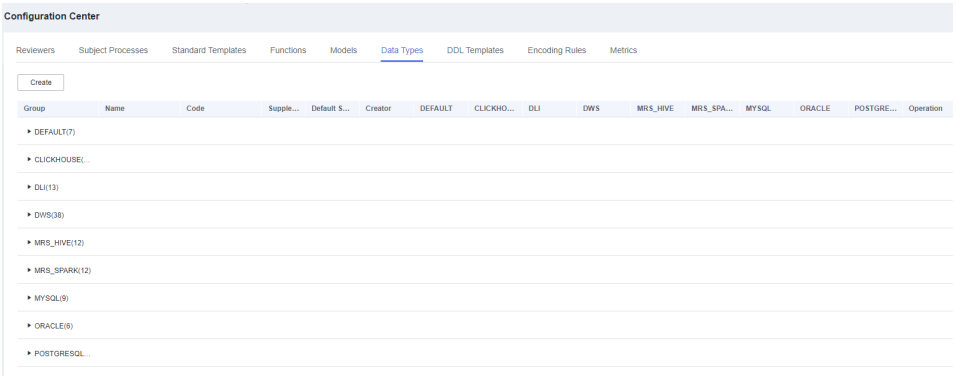
Step 1 In the navigation pane, choose **Configuration Center**. On the displayed page, click the **Data Types** tab.

Step 2 On the page displayed, you can view the data type and the data type mappings between different data sources. The type whose creator is **SYSTEM** is the default field type.

The types are described as follows:

- **DEFAULT** indicates the common data type which is used for creating a table when the data source type is not specified. For example, when you create a table of a logical model, the data type in the DEFAULT group is used.
- **DLI** indicates the data type of the table with the DLI data connection.
- **DWS** indicates the data type of the table with the DWS data connection.
- **MRS_HIVE** indicates the data type of the table with the MRS_HIVE data connection.
- **MRS_SPARK**: indicates the data type of the Hudi table with the MRS_SPARK connection.
- **POSTGRESQL**: indicates the data type of the table with the PostgreSQL connection.
- **CLICKHOUSE**: indicates the data type of the table with the ClickHouse connection.
- **MYSQL**: indicates the data type of the table with the MySQL connection.
- **ORACLE**: indicates the data type of the table with the Oracle connection.

Figure 5-14 Data Types tab page



Step 3 Manage field types.

- **Create**

  To add a field type, click **Create**. In the dialog box displayed, set the parameters and click **OK**.

Figure 5-15 Creating a field type

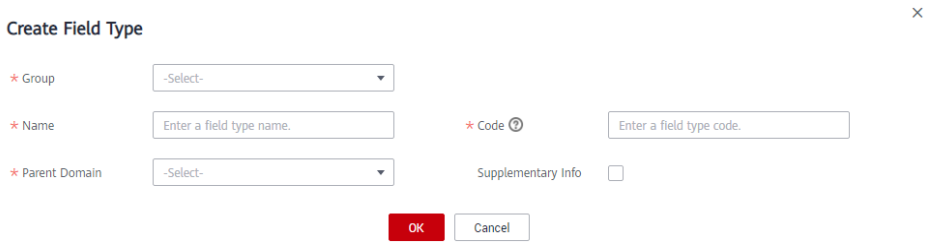


Table 5-1 Parameters for creating a field type

Parameter	Description
Group	Group that the new field type belongs to.
Name	Name of the field type to create. Field type names must start with letters. Only letters, numbers, brackets, spaces, and underscores (_) are allowed.
Code	Data type code, which must be supported by the data warehouse. The code can contain uppercase letters, underscores (_), and digits, and must start with an uppercase letter or underscore (_).
Parent Domain	Select the domain that the new field type belongs to.
Supplementary Info	You can enable this function if you want to set the data length range for some data types. For example, you can enter **(10,2)** for the DECIMAL(p,s) data type, indicating that the total number of digits in the value is 10, and the number of digits after the decimal point is 2. You can also enter **10** for the VACHAR data type, indicating that the maximum number of characters is 10.
Data Types in Data Sources	Select the data type of the mapping connection of the new field type.
DEFAULT	Data type of the default data connection that the new field type is mapped to.
CLICKHOUSE	Data type of the ClickHouse data connection that the new field type is mapped to.
DLI	Data type of the DLI data connection that the new field type is mapped to.
DWS	Data type of the DWS data connection that the new field type is mapped to.
MRS_HIVE	Data type of the MRS Hive data connection that the new field type is mapped to.
MYSQL	Data type of the MySQL data connection that the new field type is mapped to.

Parameter	Description
ORACLE	Data type of the Oracle data connection that the new field type is mapped to.
POSTGRESQL	Data type of the PostgreSQL data connection that the new field type is mapped to.

- **Edit**

  In the field type list, specify a field type and click  to edit the field type. For details on the parameters, see **Table 5-1**.

- **Delete**

  You can delete new field types. The field type whose creator is **SYSTEM** is the default field type and cannot be deleted.

  In the field type list, specify a field type and click  to delete it. Then click **OK**.

- **Reset**

  Click **Reset** at the bottom of the **Field Type** tab page to restore the default settings.

----End

DDL Templates

On the DataArts Architecture page, you can modify DDL templates of DLI views or diversified types of tables (such as DWS, DLI, POSTGRESQL, Hive, and Spark). If you need to generate DDL statements of other data sources for a created table of a certain type, you can modify the DDL template of the table based on the DDL syntax of the target data source.

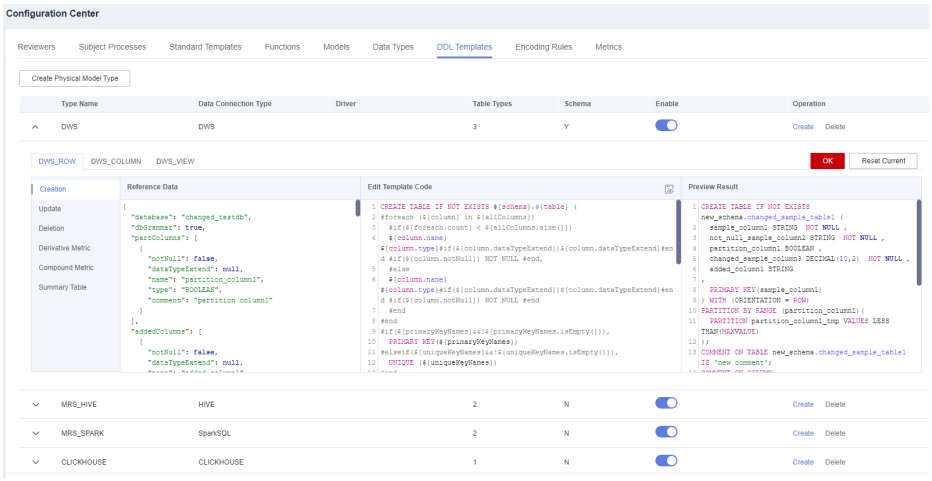
1. In the navigation pane, choose **Configuration Center**. On the displayed page, click the **DDL Templates** tab.
2. On the page displayed, you can configure DDL templates for DLI views or diversified types of tables. You can modify the DDL templates by referring to the parameter description on this page. After the modification is complete, click **OK**. Click **Reset All** to restore the default settings.

   As shown in **Figure 5-16**, the process is described as follows:

   - **Creation** allows you to view or edit a new table or a DDL template of a DLI view.
   - **Update** allows you to view or edit an updated table or a DDL template of a DLI view.
   - **Deletion** allows you to view or edit a deleted table or a DDL template of a DLI view.
   - **Derivative Metric** allows you to view or edit the SQL template of a derivative metric.
   - **Compound Metric** allows you to view or edit the SQL template of a compound metric.

- **Summary Table** allows you to view or edit the SQL template of a summary table.
- The **Reference Data** area shows an example of table details. Variables in the example define table details.
- The **Edit Code Template** area allows you to edit DDL templates. If you need to generate DDL statements for other types of databases, you can modify the DDL template based on the DDL syntax of the target data source.
- The **Preview Result** area allows you can preview the DDL statements generated based on the edited template.

Figure 5-16 DDL Templates tab page



Encoding Rules

1. In the navigation pane, choose **Configuration Center**. On the displayed page, click the **Encoding Rules** tab.
2. Manage encoding rules.
   - Add an encoding rule.

     Click **Add** above the encoding rule list. In the displayed dialog box, set required parameters, and click **OK**.

Figure 5-17 Adding an encoding rule

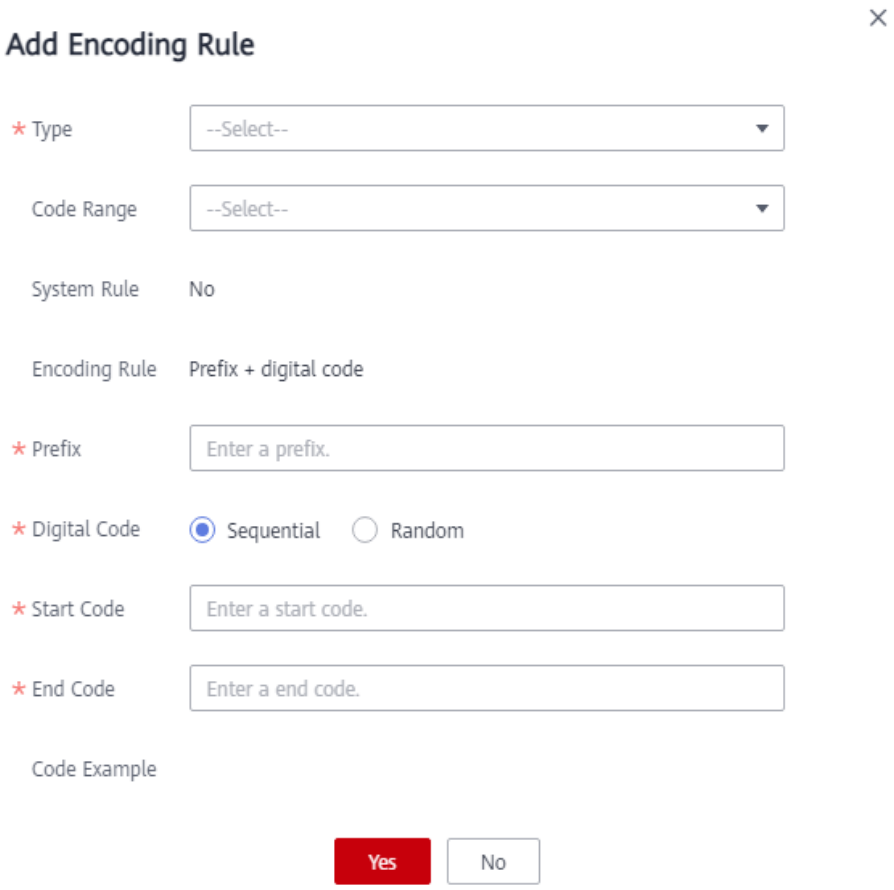


Table 5-2 Parameters for adding an encoding rule

Parameter	Description
Type	Encoding rule type. The following options are available: **Business metric**, **Logical entity**, **Logical property**, **Data standard**, and **Code Table**, and **Business Object**.
Code Range	By default, the encoding rule takes effect globally. You can select subjects, processes, lookup tables, or data standards.
System Rule	Whether this rule is a system rule. The value is **No** and cannot be changed.
Encoding Rule	The value consists of a prefix and a digit code and cannot be changed.
Prefix	The value can contain characters and digits but cannot end with a digit. It cannot be changed.
Digital Code	You can select **Sequential** or **Random**.

Parameter	Description
Start Code	Start value of the digital code range
End code	End value of the digital code range
Code Example	The configured encoding rule is displayed.

- Deleting an Encoding Rule

  Select an encoding rule and click **Delete** above the list. In the displayed dialog box, click **Yes**.

  NOTE

  The six preset encoding rules cannot be deleted, including the logical entity, data standard, logical property, business metric, code table, and business object rules.

- Editing an Encoding Rule

  Locate an encoding rule, click **Edit** in the **Operation** column, modify parameters, and click **OK**.

Metric Settings

1. In the navigation pane on the DataArts Architecture console, choose **Configuration Center**. On the displayed page, click the **Metrics** tab.
2. Manage business metrics.

   a. Create a metric.

   Click **Create** next to **Business Metric Customize Field** or + in the **Operation** column of an existing metric. Set the following parameters and click **Save**.

Figure 5-18 Creating a metric

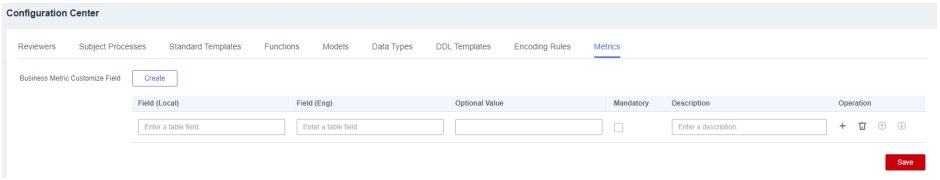


Table 5-3 Parameters for creating a metric

Parameter	Description
Field (Local)	Metric name. Enter a maximum of 100 characters.

Parameter	Description
Field (Eng)	Metric name in English. Enter a maximum of 100 characters.
Optional Value	Optional values of the custom metric for creating a business metric
Mandatory	Whether the custom metric is mandatory for creating a business metric
Description	Description of the custom metric Enter a maximum of 200 characters.

b. Adjust the metric sequence.

You can adjust the sequence of metrics by clicking the up or down arrow in the **Operation** column. You can also double-click the up or down arrow and enter a No. to move a metric to a specified row.

Figure 5-19 Adjusting the metric sequence

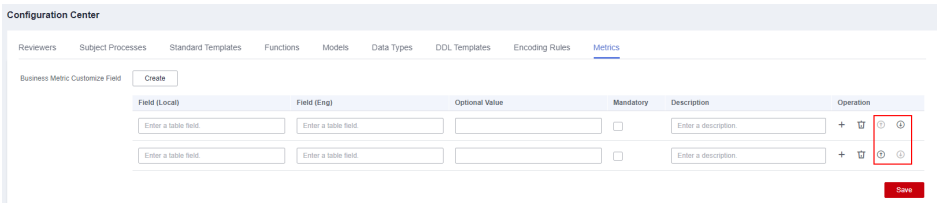
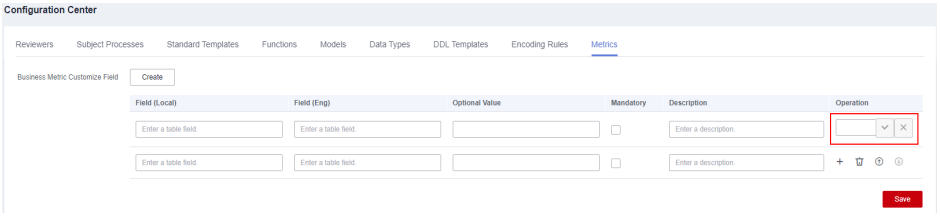


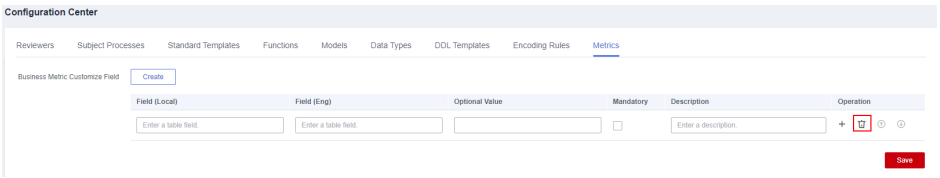
Figure 5-20 Moving a metric to a specified row



c. Delete a metric.

To delete a custom metric, click  in the **Operation** column.

Figure 5-21 Delete a metric.



3. After a custom metric is set, the metric is displayed on the page for creating a business metric and the **Basic Settings** page of the business metric.

Figure 5-22 Page for creating a business metric

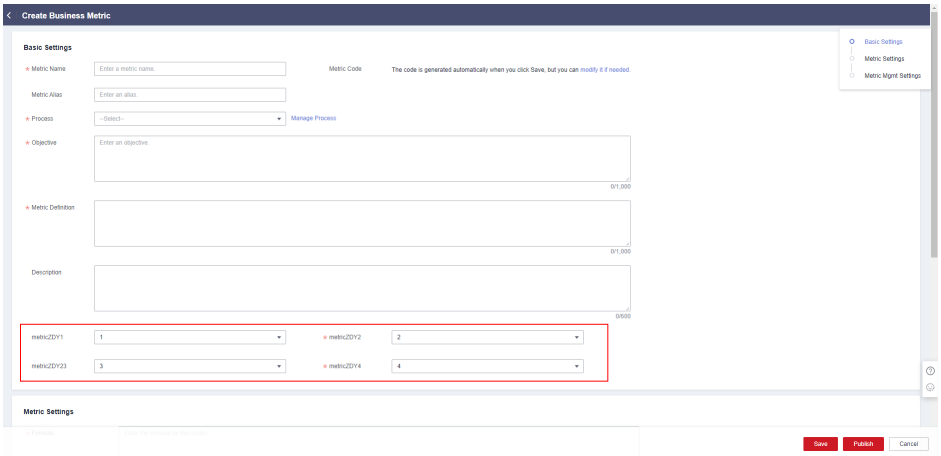
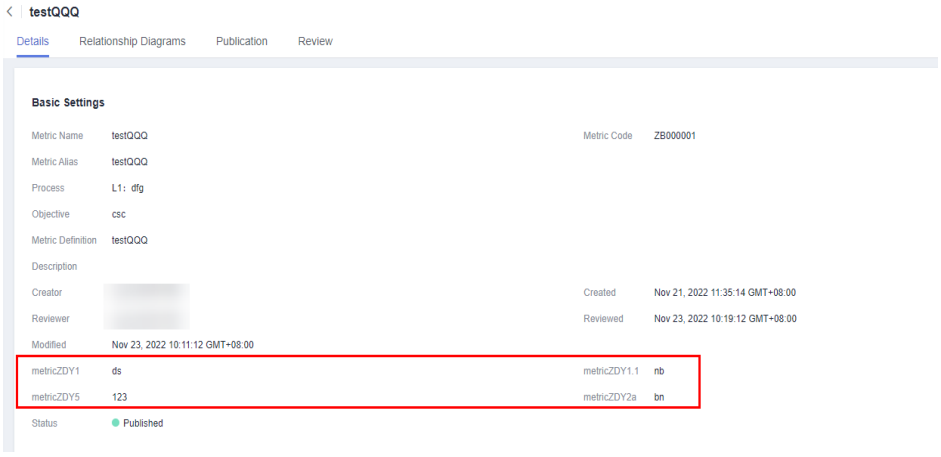


Figure 5-23 Basic Settings page of a business metric



5.4 Data Survey

5.4.1 Designing Processes

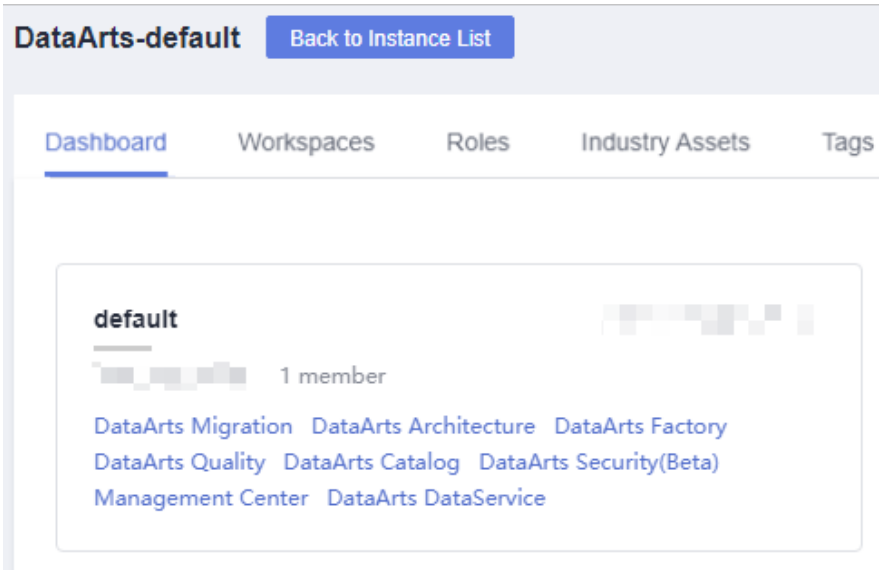
Business Process Architecture (BPA) is developed based on value streams, and is used to guide and standardize the management of requirements and ensure the efficiency of business requirement handling, analysis, and delivery. BPA prioritizes high-value requirements, which maximizes the business value, assists in business operations, and facilitates goal achievement.

Creating a Process

Design a process that consists of three to seven levels. For details about how to change the process levels, see **Process Levels**.

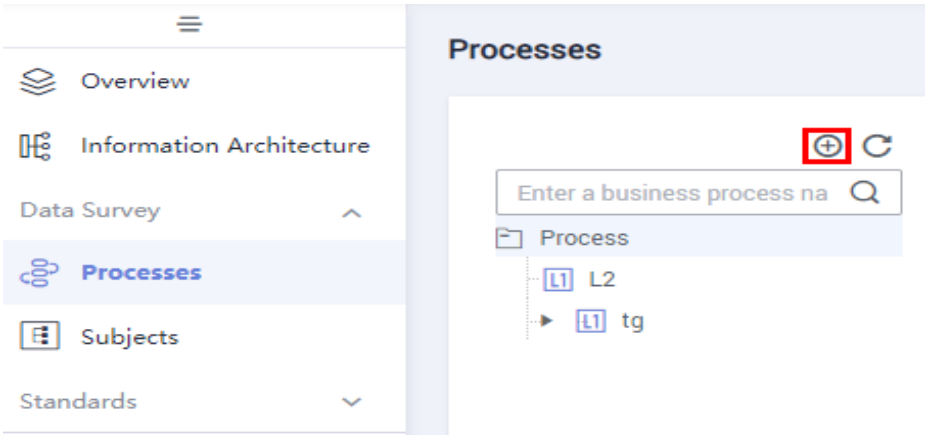
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

Figure 5-24 DataArts Architecture



2. Choose **Data Survey** > **Processes** in the left navigation bar. Click ⊕ to create a process. When creating a process for the first time, perform the operation under the root node.

Figure 5-25 Process design



3. In the dialog box displayed, set the parameters and click **OK**.

Figure 5-26 Creating a process

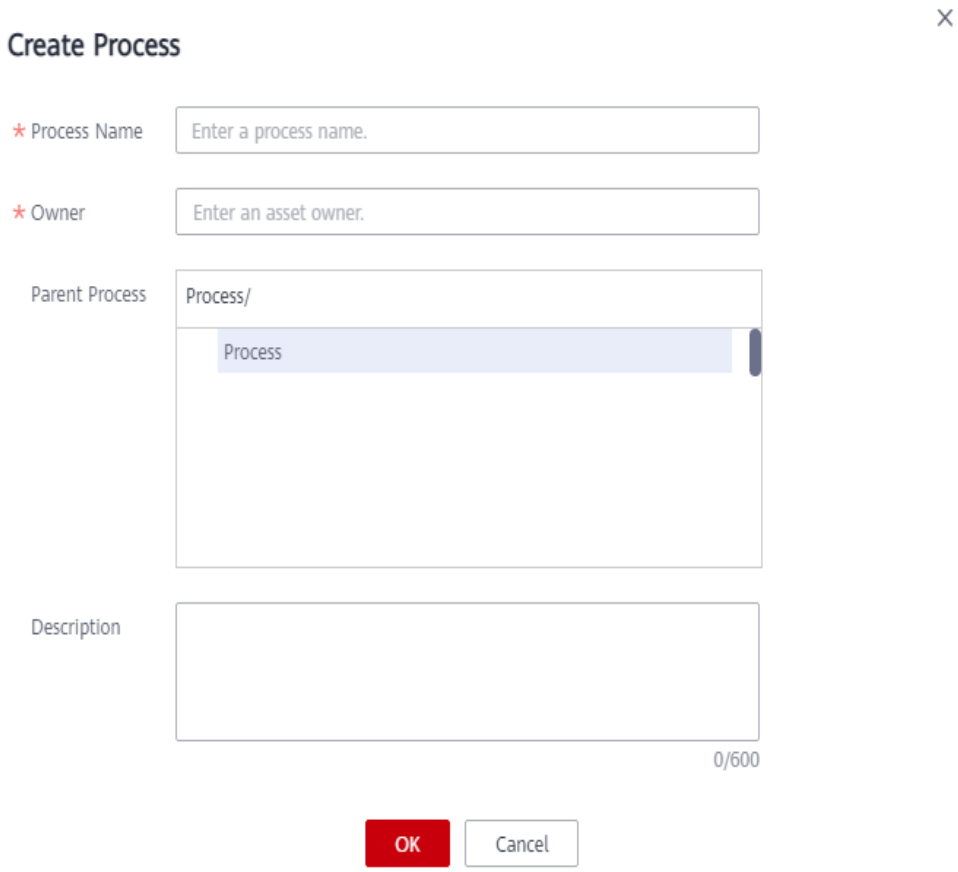


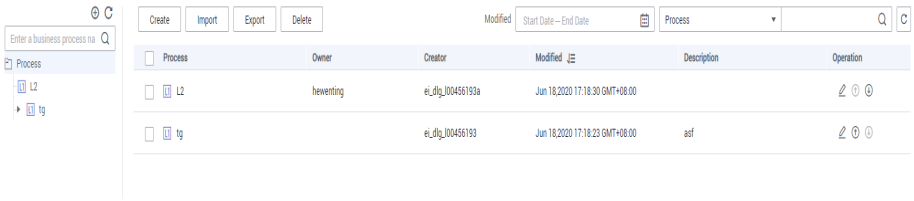
Table 5-4 Parameters for creating a process

Parameter	Description
Process	Process name. Only letters, numbers, and underscores (_) are allowed.
Owner	Process owner. You can enter the name of an owner or select an existing owner.
Parent Process	Parent process of the process
Description	A description of the process.

4. Repeat the preceding steps in sequence to create more processes or subprocesses. Generally, you must design processes from L1 to L3. The first layer is identified as L1, the second layer as L2, and the third layer as L3.

   The following figure shows an example.

Figure 5-27 Process design example



Exporting a Process

You can export the processes that have been created in DataArts Architecture to files.

Step 1 On the DataArts Architecture page, choose **Data Survey** > **Processes** in the left navigation pane.

Step 2 Click **Export** above the process list. After a few seconds, a message is displayed in the upper right corner of the page, indicating that the process is exported. You can view the export process.

NOTE

A **process** has a hierarchy. You can export only data of all levels. All processes rather than the ones you select will be exported.

----End

Importing a Process

Step 1 On the DataArts Architecture page, choose **Data Survey** > **Processes** in the left navigation pane.

Step 2 Click **Import** above the process list.

Step 3 In the dialog box displayed, set **Update Existing Data**, click **Select File**, and click **Upload**.

Figure 5-28 Importing a process

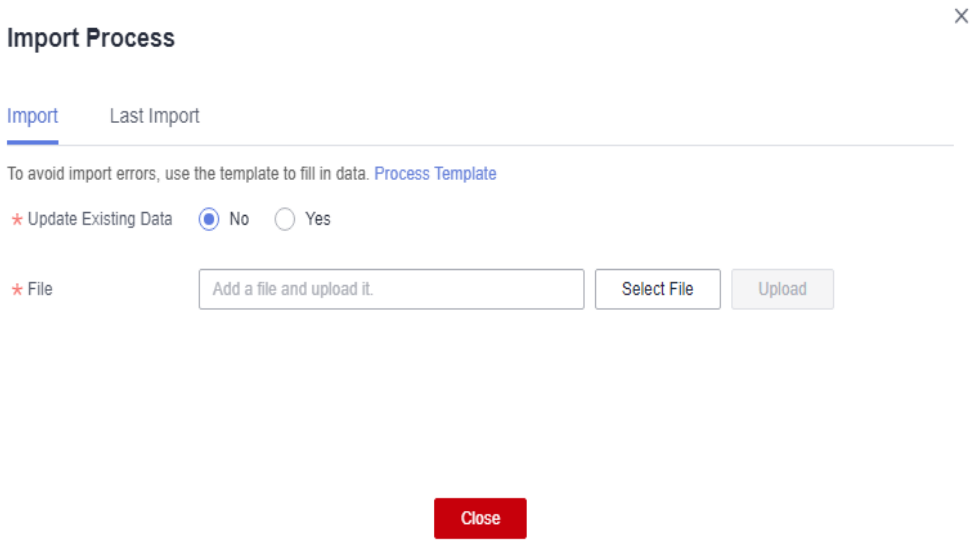


Table 5-5 Parameters for importing a process

Parameter	Description
Update Existing Data	Whether to update the existing processes of DataArts Architecture. The options are as follows: • **No**: If you select this option, the existing process will not be updated. • **Yes**: If you select this option, the existing process will be updated. During the import, only process creation and update are allowed.
Upload File	Select the file to import. You can use either of the following methods to obtain the file to import: • **Downloading the process template and fill in it** On the **Import** tab page, click **Process Template** to download the template, set related parameters in the template based on service requirements, save the settings, and upload the file. See **Table 5-6** for template parameter details. • **Exporting a process** You can export the processes created in DataArts Architecture of a DataArts Studio instance to an Excel file. Then, import the Excel file. For details, see **Exporting a Process**.

Table 5-6 describes the parameters in the downloaded template. Parameters whose names start with an asterisk (*) are mandatory, and parameters whose names do not start with an asterisk (*) are optional. One record is required for one process.

Table 5-6 Parameters in the process import template

Parameter	Description
Process	If it is a level 1 process, this field can be left blank. If it is not, this field is mandatory. If there are multiple processes, separate them with slashes (/), for example, **Integrated Product Development/Development Lifecycle**.
*Name	Process name.
*Owner	Process owner. You can enter the name of an owner or select an existing owner.
Description	A description of the process.

Step 4 The import result is displayed on the **Last Import** tab page in the **Import Process** dialog box. If the import is successful, click **Close**. If the import fails, you can view the failure cause, correct the template file, and upload it again.

----End

Deleting a Process

You can delete the processes that are no longer used. Deleted processes cannot be recovered. Exercise caution when performing this operation. If a process has subdirectories or subprocesses, you must delete the subdirectories or subprocesses first.

Step 1 On the DataArts Architecture page, choose **Data Survey** > **Processes** in the left navigation pane.

Step 2 In the process list, select the target process and click **Delete** above the process list.

Step 3 In the **Delete Process** dialog box displayed, confirm the process information and click **Yes**.

----End

5.4.2 Designing Subjects

A subject is a hierarchical architecture that classifies and defines data to help clarify data assets and specify relationships between subject areas and business objects.

You can design subjects in either of the following ways:

- **Creating and Publishing a Subject**

  Create and publish a subject.
- **Importing a Subject**

  If the subject information is complex, you are advised to import subjects in batches.
  - You can download the provided subject design template, fill in the content, and upload the file to import the subjects in batches.
  - You can export the subjects created in DataArts Architecture of a DataArts Studio instance to an Excel file. Then, import the Excel file. For details on how to export subjects, see **Exporting a Subject**.

You can search for, edit, or delete subjects.. For details, see **Managing a Subject**.

Subject Design Overview

By default, the system provides three subject levels: Subject Area Group (L1), Subject Area (L2), and Business Object (L3).

- **Subject Area Group**: used to group business domains based on scenarios
- **Subject Area**: A data domain is a dataset, in which data is of the same property.
- **Business Object** includes important information about people, events, and things that are indispensable to enterprise operations and management.

You can also customize the subject levels by referring to **Subject Processes**.

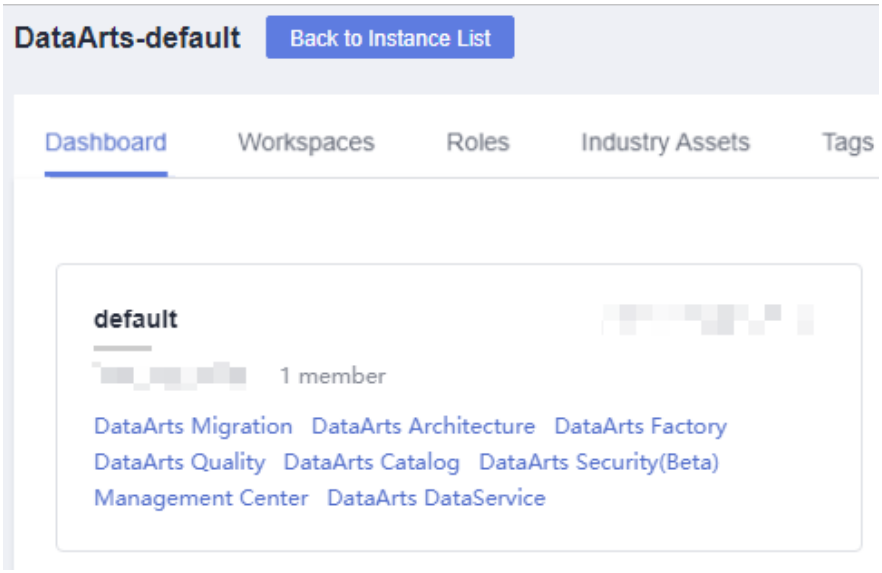
Constraints

A maximum of 500 subjects can be created in a workspace.

Creating and Publishing a Subject

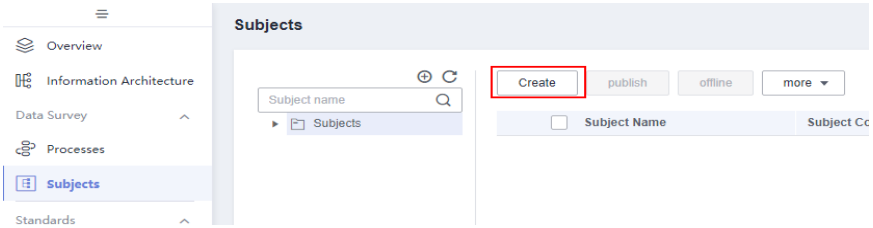
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

   Figure 5-29 DataArts Architecture

2. On the **DataArts Architecture** page, choose **Data Survey** > **Subjects** in the left navigation bar. On the page displayed, click **Create** in the upper left corner.

   Figure 5-30 Designing a subject

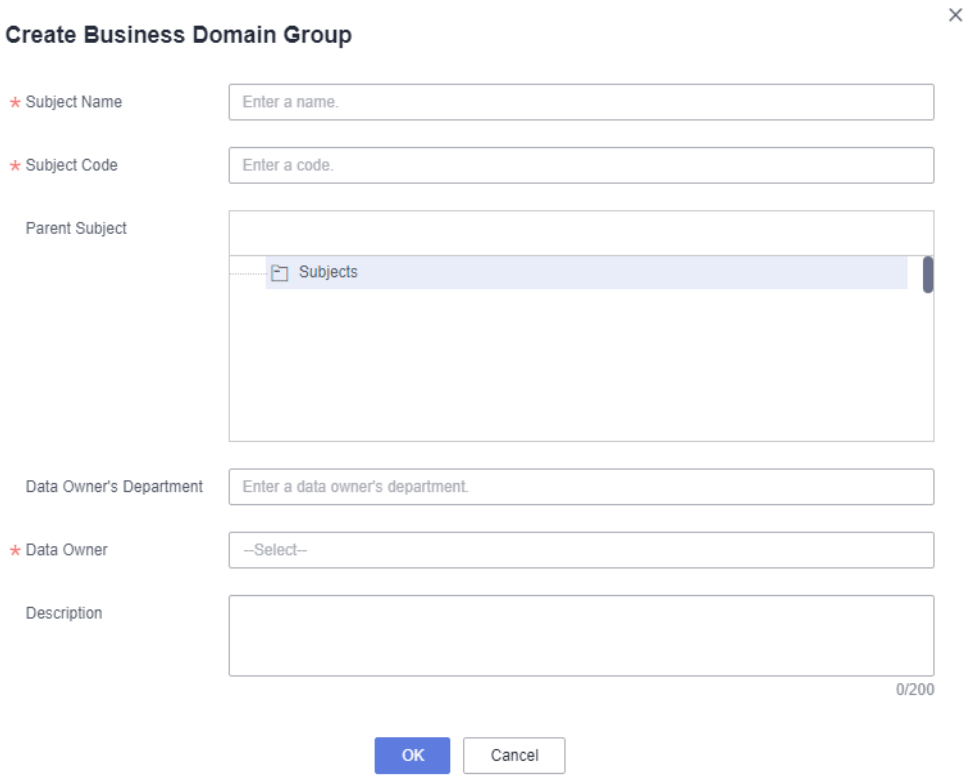
3. In the dialog box displayed, set the parameters and click **OK**.

   Table 5-7 Parameters for creating a subject area group

Parameter	Description
* Subject Name	The following characters are not allowed: / \ < >.

Parameter	Description
* Subject Code	The code of the subject area group to create. Only letters, digits, spaces, underscores (_), hyphens (-), parentheses, and ampersands (&) are allowed.
Parent Subject	Parent subject of the subject area group
Data Owner's Department	The department that the data owner belongs to.
* Data Owner	Select a data owner from the drop-down list box. You can select multiple data owners or enter custom data owners.
Description	A description of the subject area group to create.

Figure 5-31 Creating a subject

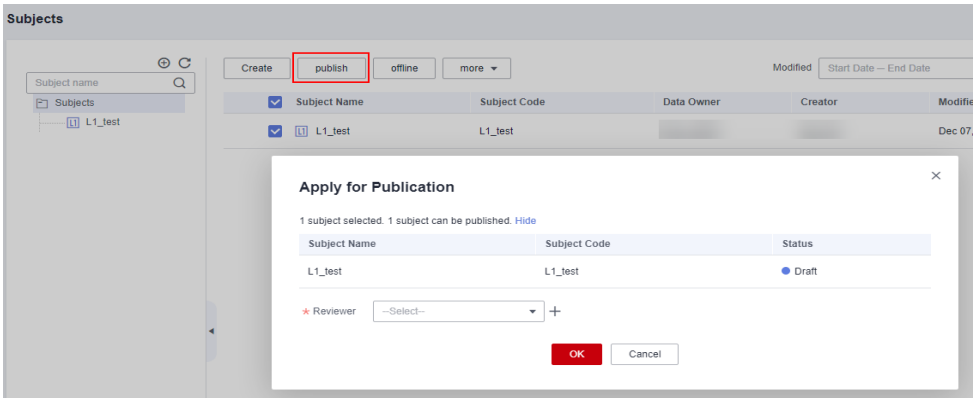


4. Select the created subject area group and click **Publish**. In the displayed dialog box, select a reviewer and click **OK**. After the application is approved, the **Subjects** page is displayed. You can view the created subject area group in the list, and the status of the subject area group is **Published**. Only published subject area groups can be used.

   NOTE

   If you have been added as a reviewer, you can select **Auto-review** and click **OK**. After the application is approved, the subject area group status changes to **Published**.

Figure 5-32 Publishing a subject



5. You can create multiple subjects in a subject. Note that a subject can be published only if its upper-layer subjects have been published.

   NOTE

   When you are creating a L3 subject, that is, a business object, parameter **Subject Code** is displayed in the **Create Business Object** dialog box. You can select **Auto Generate** or **Custom**.

   - **Auto Generate**: A code is automatically generated based on the **encoding rule** in the Configuration Center.
   - **Custom**: Enter a code.

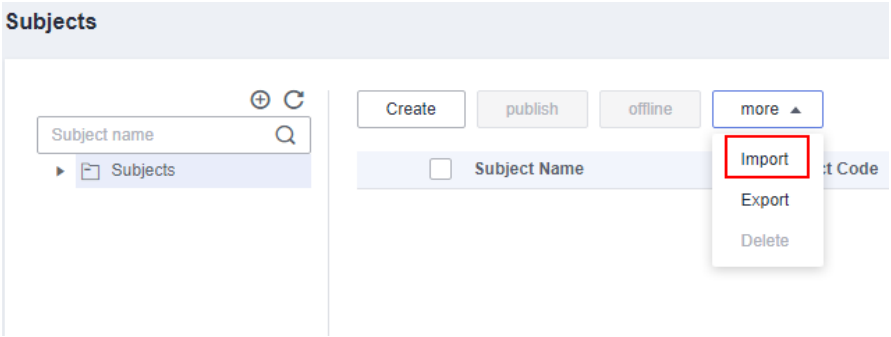
   The number of subject levels is defined by users on the **Subject Levels** tab page on the **Configuration Center** page. By default, there are three levels in the system, Subject Area Group (L1), Subject Area (L2), and Business Object (L3).

Importing a Subject

Step 1 On the DataArts Architecture page, choose **Data Survey** > **Subjects** in the left navigation pane.

Step 2 Click **More** above the subject list and select **Import**.

Figure 5-33 Importing a subject



Step 3 In the dialog box displayed, set **Update Existing Data**, click **Select File**, and click **Upload**.

Figure 5-34 Importing a subject

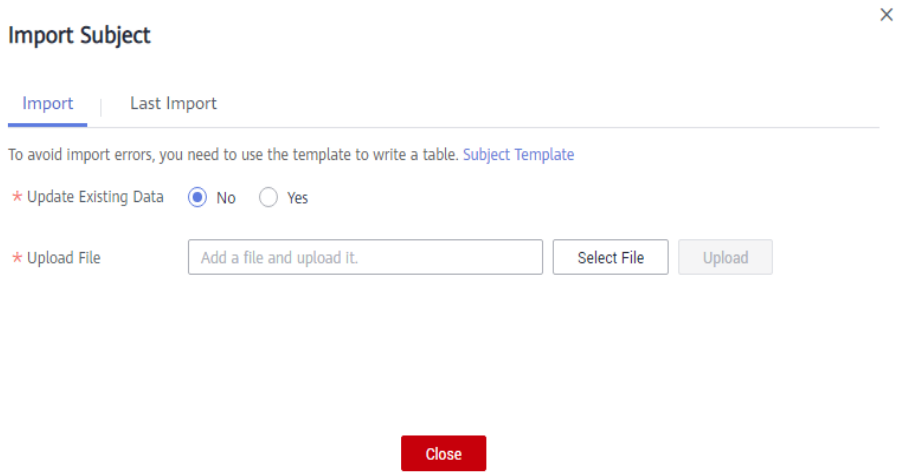


Table 5-8 Parameters for importing subjects

Parameter	Description
Update Existing Data	Whether to update existing subject information (subject area group, subject area, or business object) during the import. When a subject is imported, the system checks whether the subject exists according to its code. • **No**: If you select this option, the subject information will not be updated. • **Yes**: If you select this option, the subject information will be updated. During the import, only subject creation and update are allowed.
Upload File	Select the file to import. You can use either of the following methods to obtain the file to import: • **Downloading the subject import template and fill in it** In the **Import Subject** dialog box, click **Subject Template** to download the template, fill in the content, and save the settings. See **Table 5-9** for template parameter details. • **Exporting subjects to files** You can export the subjects created in DataArts Architecture of a DataArts Studio instance to an Excel file. Then, import the Excel file. See **Exporting a Subject** for details.

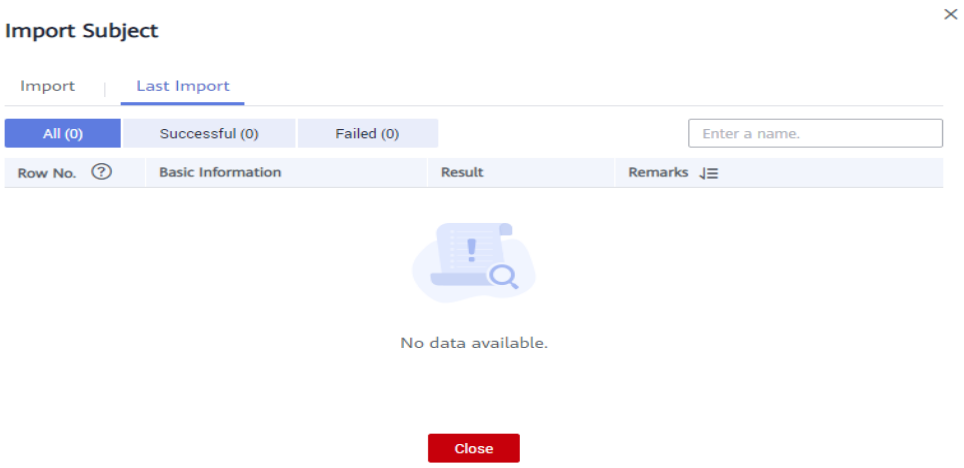
Table 5-9 describes the parameters in the downloaded template. Parameters whose names start with an asterisk (*) are mandatory, and other parameters are optional. Enter the information about a subject in a line.

Table 5-9 Parameters

Parameter	Description
Parent Subject	Encoding path of the upper-level subject, which is separated by slashes (/).
*Name	The following characters are not allowed: / \ < >.
*Code	Code of the subject to create. Only letters, digits, spaces, underscores (_), hyphens (-), parentheses, and ampersands (&) are allowed.
Alias	Alias of the subject.
Description	A description of the subject. This parameter is mandatory for the lowest-level subject. You must add the description of the lowest-level subject in the file to be imported.
Data Owner's Department	The department that the data owner belongs to. This parameter is mandatory for the lowest-level subject. You must add the department of the owner of the lowest-level subject in the file to be imported.
Data Owner	The owner of the data. Multiple owners are supported. Separate owner names with commas (,)

Step 4 View the import result on the **Last Import** tab page. If the import is successful, click **Close**. If the import fails, you can view the failure cause, correct the template file, and upload it again.

Figure 5-35 Last Import tab page



----End

Exporting a Subject

Step 1 On the DataArts Architecture page, choose **Data Survey** > **Subjects** in the left navigation pane.

Step 2 Click **More** above the subject list and select **Export** to export the subjects to an Excel file. Then, import the Excel file.

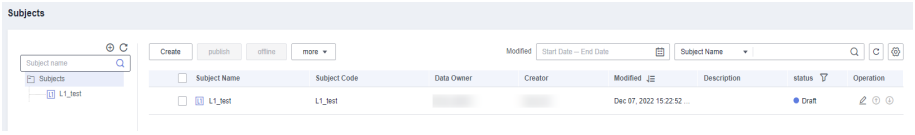
NOTE

A subject or process has a hierarchy. You can export only data of all levels.

----End

Managing a Subject

Figure 5-36 Subject design area



- Search

  You can enter a keyword in the search box to search for all subjects in the public workspace.

- Edit

  Locate a subject in the list and click  in the **Operation** column to edit the subject. To make a published subject take effect after you have edited it, select the draft and publish it.

- Delete

  Select a subject in the list and click **More** and select **Delete** above the list to delete the subject.

- Move Up/Down

  Locate a subject in the list and click  or  in the **Operation** column to move down or up the subject.

5.5 Standards Design

5.5.1 Creating Lookup Tables

A lookup table is also called a data dictionary table. It consists of enumerable data names and codes and stores the relationships between them. A lookup table provides the following functions:

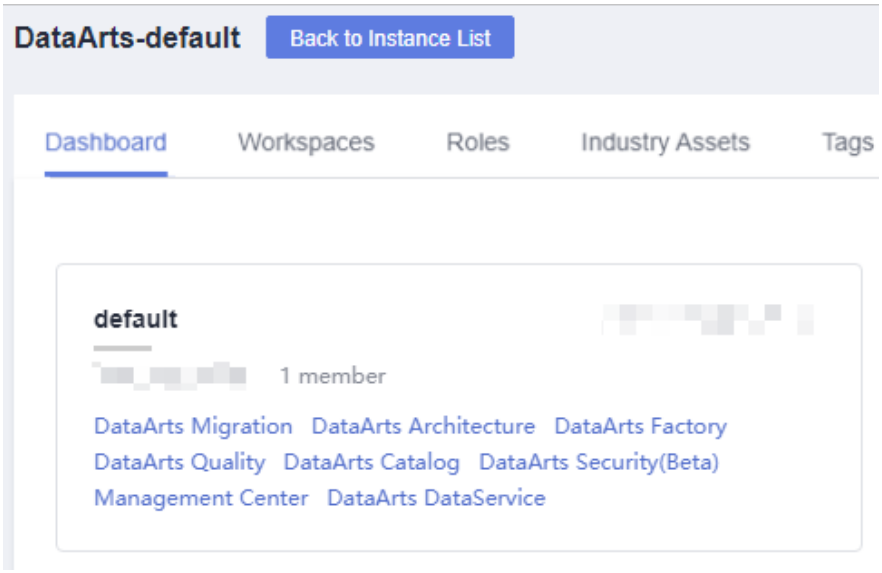
- Standardizes business data and supplements mapping fields during data cleansing.
- Monitors the value range of business data during data quality monitoring.
- Enumerates dimensions during dimensional modeling.

Creating and Publishing a Lookup Table

Manually create a lookup table. You can also add table records after creating a lookup table. For details, see **Filling in a Lookup Table**.

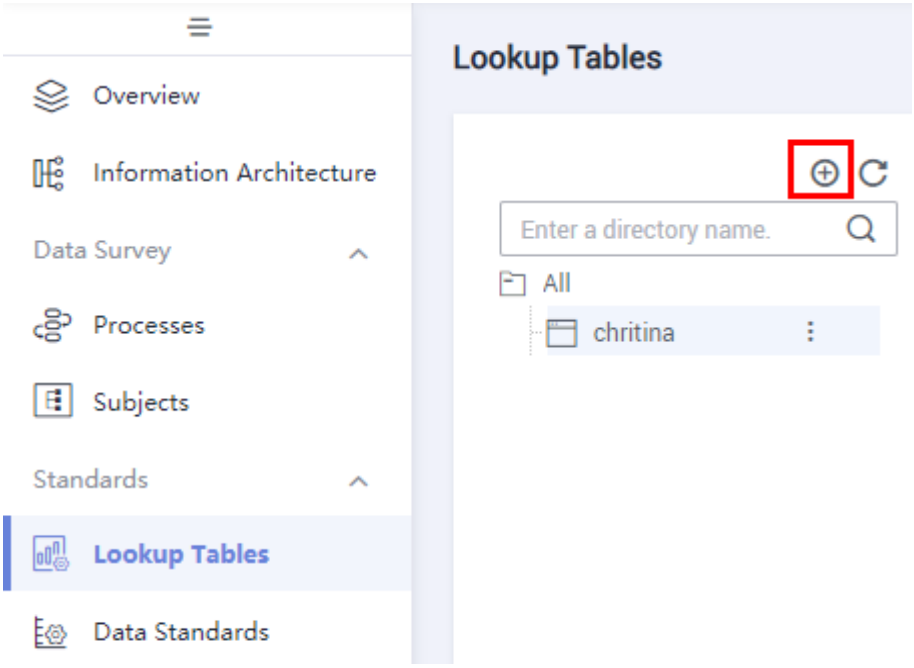
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

   Figure 5-37 DataArts Architecture

2. On the DataArts Architecture page, choose **Standards** > **Lookup Tables** in the left navigation pane.
3. Select a directory from the directory tree on the **Lookup Tables** page, and then click ⊕ to create a directory under the selected directory. When creating a directory for the first time, you can create a directory under the root directory.

Figure 5-38 Lookup Tables page



4. In the dialog box displayed, set the parameters and click **OK**.

Figure 5-39 Create Directory dialog box

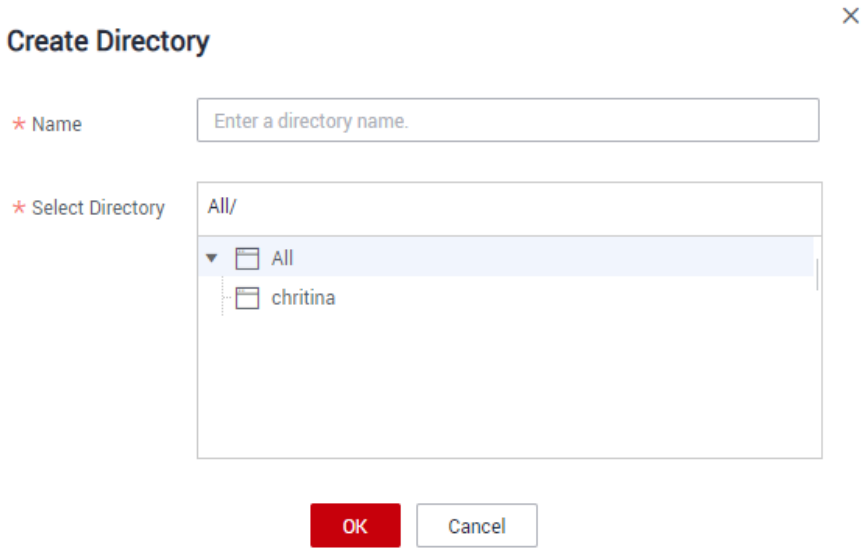


Table 5-10 Directory parameters

Parameter	Description
*Name	The following characters are not allowed: / \ . < >.
*Select Directory	Select an existing directory, and create a subdirectory under it.

5. Select the directory you created in the directory tree and click **Add** to create a lookup table.
6. On the **Create Lookup Table** page displayed, configure the parameters.

   In the **Table Details** area, set the parameters.

Figure 5-40 Table Details area

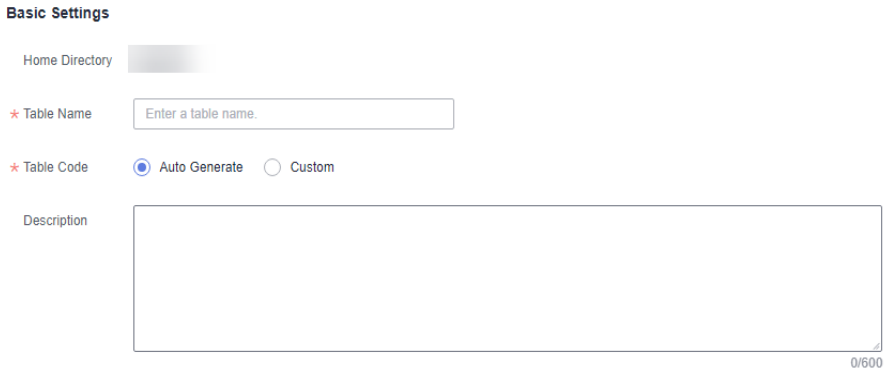
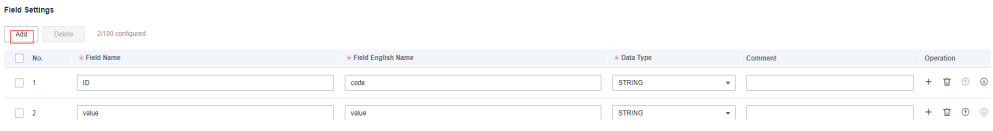


Table 5-11 Parameters

Parameter	Description
*Table Name	Name of the lookup table to create. The value can contain only letters, digits, brackets, commas (,), and the following special characters: +-#_[]/. It must start with a letter.
*Table Code	The code of the lookup table to create. You can select **Auto Generate** or **Custom** (enter a custom code). The lookup table name must start with letters. Only letters, digits, and underscores (_) are allowed.
Description	A description of the lookup table. Up to 600 characters are supported.

In the **Field Inputs** area, click **Add** or + to add new fields, and click  to delete unnecessary fields.

Figure 5-41 Field Inputs area



7. Click **Publish**. In the **Apply for Publication** dialog box displayed, select a reviewer and click **OK**. After the application is approved, the **Lookup Tables**

page is displayed. You can view the created lookup table in the list, and the status of the table is **Published**. Only published lookup tables can be used.

NOTE

If you have been added as a reviewer, you can select **Auto-review** and click **OK**. After the application is approved, the lookup table status changes to **Published**.

Filling in a Lookup Table

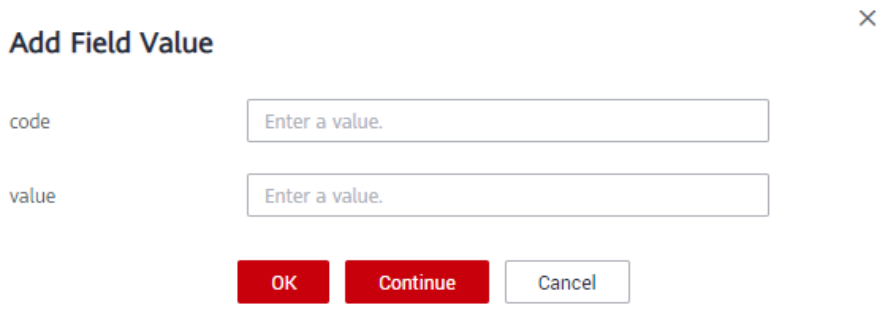
Input values in the created lookup tables.

Step 1 On the DataArts Architecture page, choose **Standards** > **Lookup Tables** in the left navigation pane.

Step 2 In the list of lookup tables, find the target table and choose **More** > **Manage Value** in the **Operation** column.

Step 3 On the page displayed, click **Add**. In the dialog box displayed, set the parameters.

Figure 5-42 Inputting a value



Step 4 Click **OK**. You can also click **Continue** to add more records.

----End

Importing a Lookup

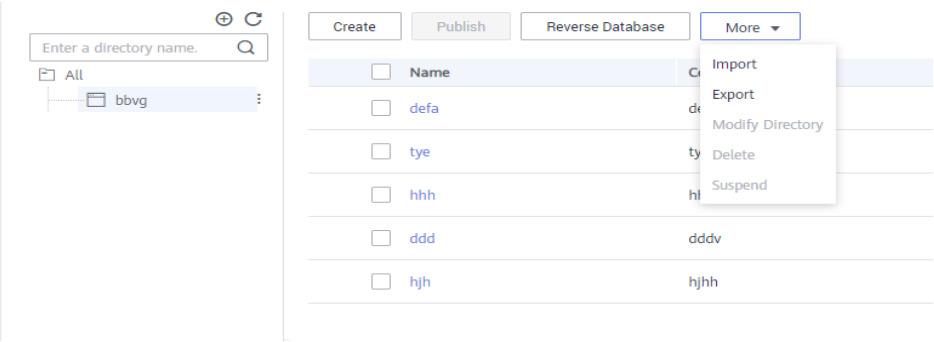
When importing a lookup table, ensure that the table name contains a maximum of 32 characters.

You can import a new lookup table or import lookup table records in batches to an existing lookup table. If you have a large number of lookup table records, you are advised to import them in batches.

Step 1 On the DataArts Architecture page, choose **Standards** > **Lookup Tables** in the left navigation pane.

Step 2 On the page displayed, select a directory, and choose **More** > **Import**. You can also right-click the selected directory and choose **Import**.

Figure 5-43 Lookup Tables page



Step 3 In the **Import Lookup Table** dialog box displayed, set the parameters, and click **Upload**.

Figure 5-44 Import Lookup Table dialog box

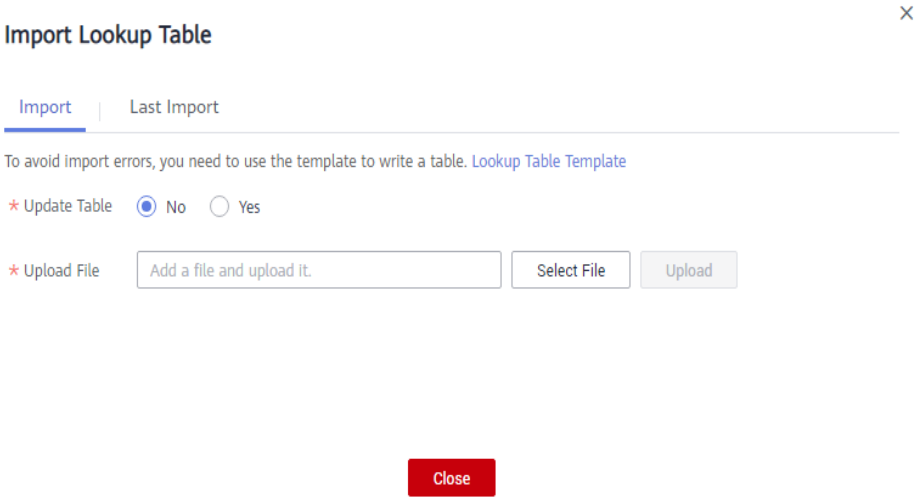


Table 5-12 Parameters for importing a lookup table

Parameter	Description
*Update Table	Whether to update the existing lookup table. When a lookup table is imported, the system checks whether the lookup table exists according to its code. The options are as follows: • **No**: If you select this option, the existing lookup table will not be updated. • **Yes**: If you select this option, the existing lookup table will be updated. If a lookup table is in the **Published** state, you must publish the lookup table again after updating it so that the updated lookup table can take effect. The import can create a lookup table or update an existing lookup table. It will not delete a lookup table.

Parameter	Description
*Upload File	Select the file to import. You can use either of the following methods to obtain the file to import: • **Downloading the lookup table template and fill in it** In the **Import Lookup Table** dialog box, click **Lookup Table Template** to download the template, fill in the content, and save the settings. See **Table 5-13** for template parameter details. Instructions for filling in the lookup table template: – Parameters whose names start with an asterisk (*) are mandatory, and other parameters are optional. – Multiple fields can be added to a lookup table. – To import multiple lookup tables, you can add multiple sheets to the template file. The sheet name is the corresponding lookup table code. – If the name of a lookup table already exists and **Update Table** is set to **Yes**, the existing lookup table will be updated during the import. – If the table name does not exist, a lookup table with that name is created during the import. • **Exporting lookup tables to files** You can export the lookup tables created in DataArts Architecture of a DataArts Studio instance to an Excel file. Then, import the Excel file. For details on how to export lookup tables, see **Managing a Lookup Table**.

Table 5-13 Parameters

Parameter	Description
Directory	The directory that a lookup table belongs to. Multi-level directories are separated with slashes (/), for example, **dir01/dir02**.
*Table Name	The name of the lookup table to create. It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_
*Table Code	The code of the lookup table to create. Only letters, numbers, and underscores (_) are allowed. A table code must start with a letter.
Table Description	A description of the lookup table. Up to 600 characters are supported.
*Field Name	The name of a field. Field names must start with letters. Only letters, numbers, spaces, and the following special characters are allowed: ()-_

Parameter	Description
*Field Code	The code of a field. Only letters, numbers, and underscores (_) are allowed. A field code must start with a letter.
*Field Data Type	The possible values are **STRING**, **BIGINT**, **DOUBLE**, **TIMESTAMP**, **DATE**, **BOOLEAN**, and **DECIMAL**.
Field Description	The supplementary information about a field. Up to 600 characters are supported.
Generate Standard	• **true** indicates to generate a data standard. • **false** indicates not to generate a data standard. The default value is **false**. Note: To enable automatic generation of the data standard, choose **Configuration Center** in the navigation pane, click the **Standard Templates** tab, and select **Lookup table**.

If the lookup table records need to be imported, create a sheet named after the lookup table in the template and add table fields to the sheet. Each field occupies a column. The column name includes the code and value. Enter the lookup table values to be imported. If the template contains a sheet named after the lookup table, you do not need to create the sheet. You can directly enter the table values to be imported in the sheet.

Step 4 View the import result on the **Last Import** tab page. If the import is successful, click **Close**. If the import fails, you can view the failure cause, correct the template file, and upload it again.

----End

Importing a Lookup Table Through a Reverse Database

With reverse databases, you can import one or more created database tables from other data sources into a lookup table directory to turn them into lookup tables.

Step 1 On the DataArts Architecture page, choose **Standards** > **Lookup Tables** in the left navigation pane.

Step 2 On the page displayed, select a directory and click **Reverse Database** above the lookup table list.

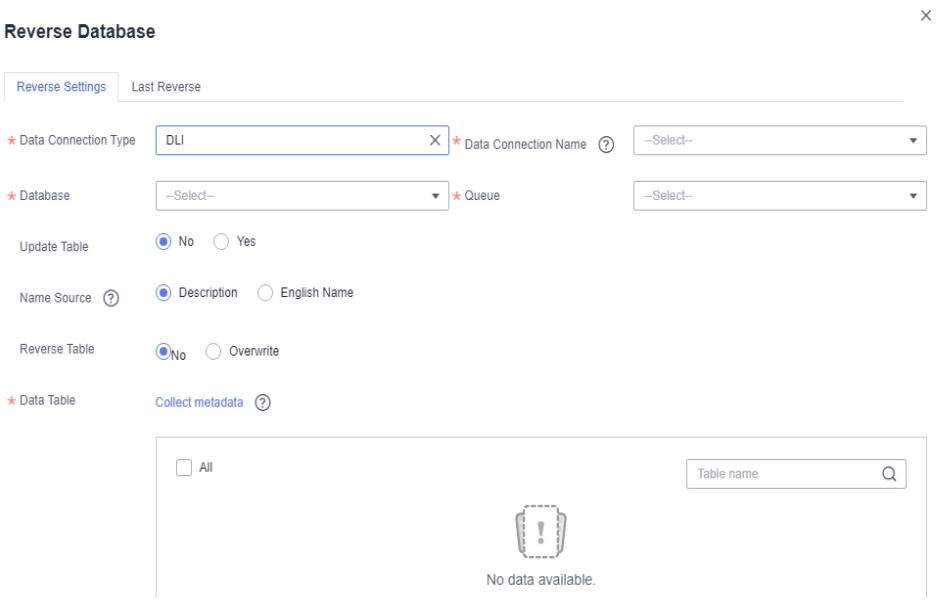
Step 3 In the dialog box displayed, set the parameters and click **OK**.

Table 5-14 Parameters for reversing a database

Parameter	Description
*Data Connection Type	The data connection types supported by the reverse database are displayed in the drop-down list box. Select the required data connection type.

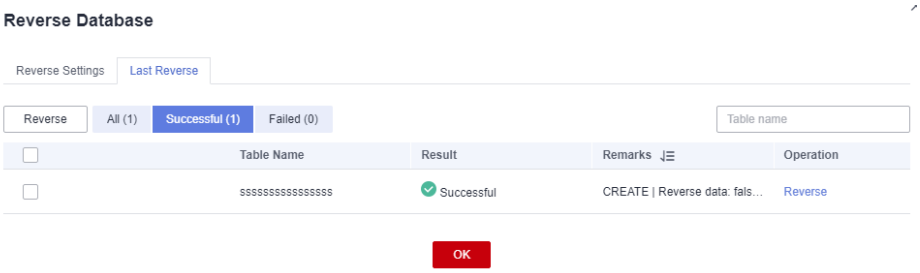
Parameter	Description
*Data Connection	Select a data connection. If you want to reverse a database from other data sources to a lookup table directory, you must create a data connection in Management Center to connect to the data source. For details on how to create data connections, see **Managing Data Connections**.
*Database	The name of the database. Select a database from the drop-down list box.
*Schema	Select a value from the drop-down list box. This parameter is available only for DWS and PostgreSQL tables.
Queue	DLI queue. This parameter is available only when **Data Connection Type** is set to **DLI**.
Update Table	When **Yes** is selected, if the name of the reversed table is the same as that of an existing table in the lookup table list, the existing table is updated.
Reverse Table	• **No**: If you select this option, tables are imported to the lookup table directory but table data is not imported during database reverse. After reversing a database, you can add records to the lookup table. Refer to **Filling in a Lookup Table** for details. • **Overwrite**: If you select this option, tables are imported to the lookup table directory and table data is imported as well during database reverse.
*Data Table	You can select one or more data tables to import.

Figure 5-45 Reverse Database dialog box



Step 4 You can view the result on the **Last Reverse** tab page. If the reverse operation is successful, click **Close**. If the reverse operation fails, you can view the failure cause. After the fault is rectified, select the table again and click **Reverse** to retry.

Figure 5-46 Last Reverse tab page



----End

Exporting a Lookup Table

When exporting a lookup table, ensure that the table name contains a maximum of 32 characters.

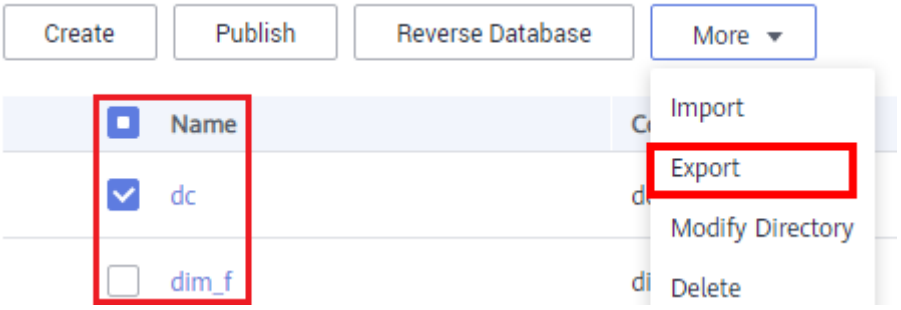
Step 1 On the DataArts Architecture page, choose **Standards** > **Lookup Tables** in the left navigation pane.

Step 2 Export a lookup table.

- **Export a single lookup table.**

  In the lookup table list, select the target lookup table and choose **More** > **Export**.

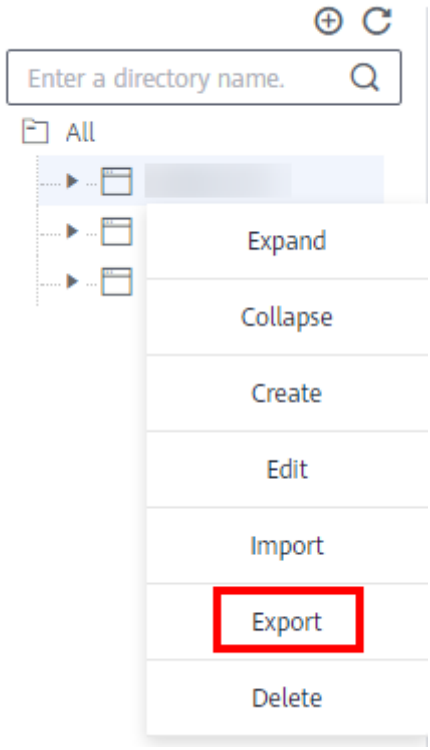
  Figure 5-47 Lookup table list

- **Export all tables in the list.**

  Right-click a directory in the directory tree and choose **Export**.

Figure 5-48 Directories storing exported lookup tables



----End

Deleting a Lookup Table

Deleted lookup tables cannot be recovered. Exercise caution when performing this operation. A lookup table in publishing review, published, or suspension review state cannot be deleted.

Step 1 On the DataArts Architecture page, choose **Standards** > **Lookup Tables** in the left navigation pane.

Step 2 In the lookup table list, select the target lookup table and choose **More** > **Delete** above the list.

Step 3 In the dialog box displayed, click **Yes**.

----End

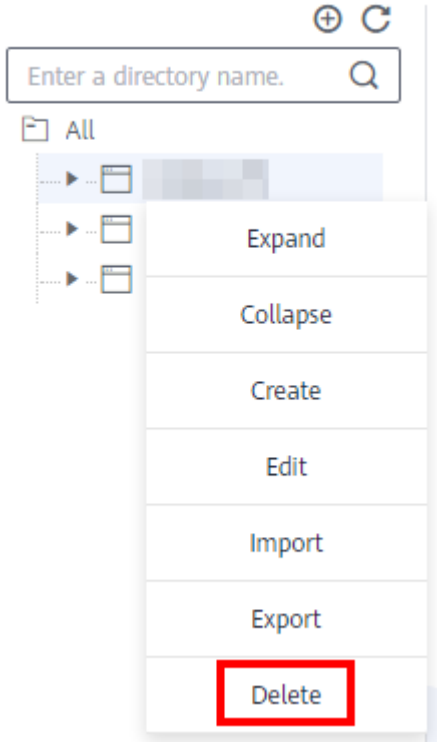
Deleting a Lookup Table Directory

A directory or its subdirectories that contain a lookup table cannot be deleted. You must delete the lookup table before deleting the directory.

Step 1 On the DataArts Architecture page, choose **Standards** > **Lookup Tables** in the left navigation pane.

Step 2 Right-click a directory in the directory tree and choose **Delete**.

Figure 5-49 Managing lookup table directories



Step 3 In the dialog box displayed, click **Yes**.

----End

Managing a Lookup Table

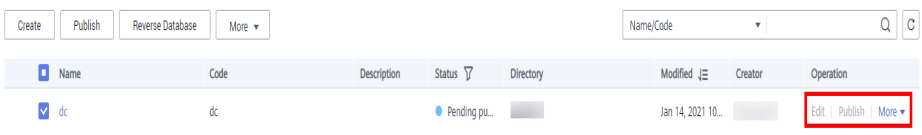
After a lookup table is created, you can search for, edit, or delete it.

On the DataArts Architecture page, choose **Standards** > **Lookup Tables** in the left navigation pane. You can manage the lookup tables as required.

NOTE

- The lookup tables created in the **public workspace** can be queried in a common workspace, but the lookup tables created in a common workspace cannot be queried in the **public workspace**.
- A common workspace has the edit permission of only the lookup tables and directories created in the same workspace, and can view indexes in the **public workspace** rather than perform any operation on the lookup tables and directories in the **public workspace**.

Figure 5-50 Managing lookup tables



- **Edit**

In the lookup table list, select a table you want to edit and click **Edit** in the **Operation** column.

- **Publish**

  In the lookup table list, click **Publish** in a row containing a table in the **Draft** or **Rejected** state, select a reviewer in the dialog box displayed, and click **OK**. After the application is approved, the lookup table is published.

- **Suspend**

  In the lookup table list, locate a published lookup table you want to suspend, click **More** in the **Operation** column, and select **Suspend** from the drop-down list. In the displayed dialog box, select a reviewer and click **OK**. After the application is approved, the lookup table is suspended.

- **Manage Value**

  In the lookup table list, locate a lookup table, click **More** in the **Operation** column, and select **Manage Value** from the drop-down list. Then you can edit the value of each field.

- **View History**

  In the lookup table list, locate a lookup table, click **More** in the **Operation** column, and select **View History** from the drop-down list. Then you can view the publish history and changes of the lookup table, and compare different versions of it.

5.5.2 Creating Data Standards

Data standards describe data meanings and business rules that are stipulated and commonly recognized by enterprises and that those enterprises must comply with.

A data standard, also called a data element, is the smallest unit of data used. It cannot be further divided. A data standard is a data unit whose definition, identifiers, representations, and allowed values are specified by a group of properties. You can associate data standards with databases of a wide range of businesses. The identifier, data type, expression format, and value range are the basis of data exchange. They are used to describe field metadata of a table and standardize data information stored in a field.

This topic describes how to create a data standard. A created data standard can be associated with fields in a business table created during ER modeling, ensuring that fields in the business table comply with the specified data standards.

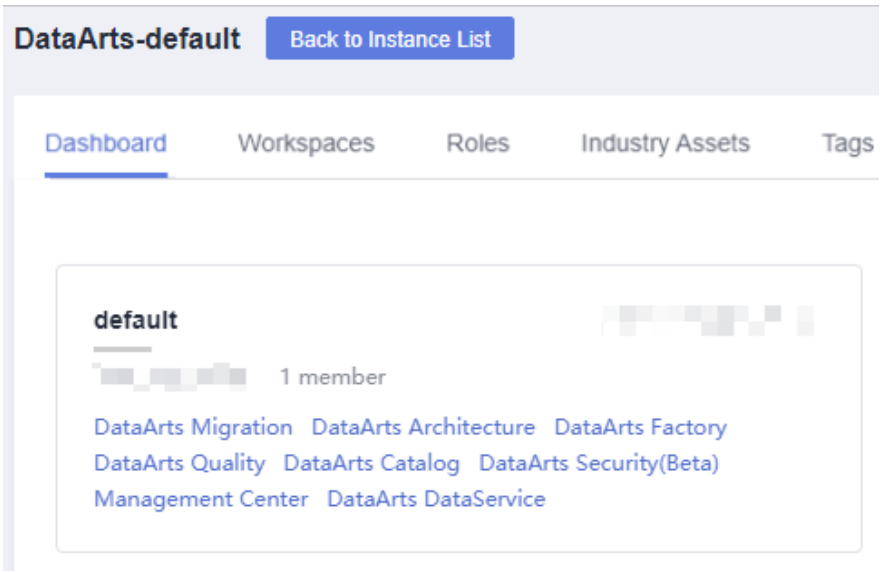
Constraints

A maximum of 500 data standard directories and 20,000 standards can be created in a workspace.

Creating a Data Standard Directory

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

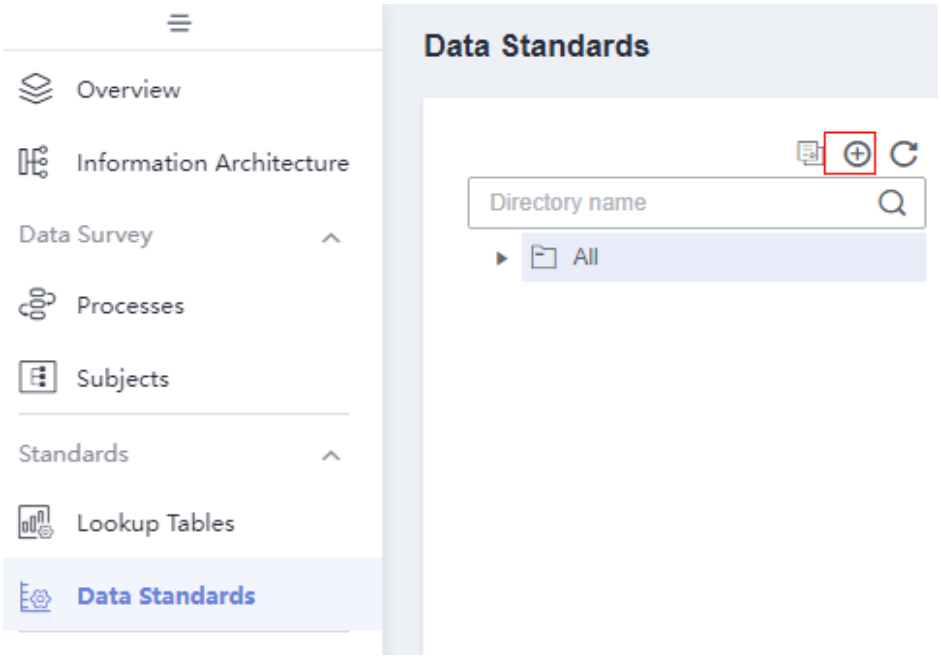
Figure 5-51 DataArts Architecture



2. On the DataArts Architecture page, choose **Standards** > **Data Standards** in the left navigation pane.
3. When you access the **Data Standards** page for the first time, the page where you can customize a standard template is displayed. Select the required options for **Optional**, add custom items, and click **Update**.

   After saving the template settings, you can modify it on the **Standard Templates** tab page of **Configuration Center**. For details, see **Standard Templates**. When creating a data standard, you must set the selected options in the template.
4. On the **Data Standards** page, select a directory and click ⊕ to create a directory under the selected one. When creating a directory for the first time, you can create a directory under the root directory.

Figure 5-52 Data Standards page



5. In the dialog box displayed, set the parameters and click **OK**.

Figure 5-53 Create Directory dialog page

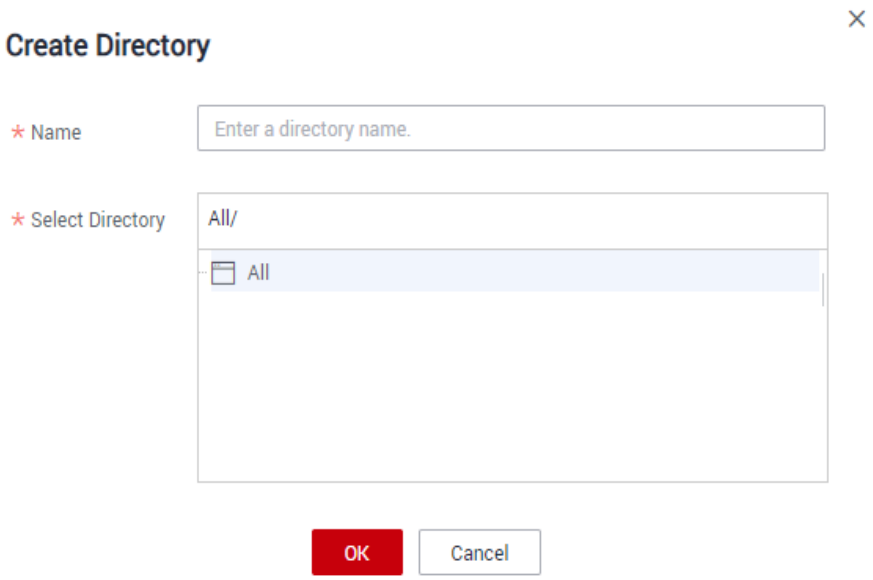


Table 5-15 Parameters for creating directories

Parameter	Description
*Name	The following characters are not allowed: / \ . < >.

Parameter	Description
*Select Directory	Select an existing directory, and create a subdirectory under it.

Click to refresh the directories.

Click to refresh the directories and synchronize subject directories to data standard directories.

NOTE

- Before synchronizing subject directories, check whether there are released subjects in the current workspace. If there are no released subjects, an error will occur during the synchronization.
- A maximum of five levels of subject directories can be synchronized to data standard directories. Subject directories beyond this range will not be synchronized. The number of directories after the synchronization cannot exceed the upper limit (generally 500). Otherwise, an error will occur and the synchronization will be canceled. Before a synchronization, the system checks for and deletes empty data standard directories. These directories and their subdirectories do not contain any data standard.
- The synchronized subject directories are displayed as L1 to L5 icons, and the existing data standard directories are displayed as their original icons.

Creating a Data Standard

Step 1 On the **Data Standards** page, select a directory and click **Create**.

Step 2 Set the parameters based on **Table 5-16** and click **Publish**.

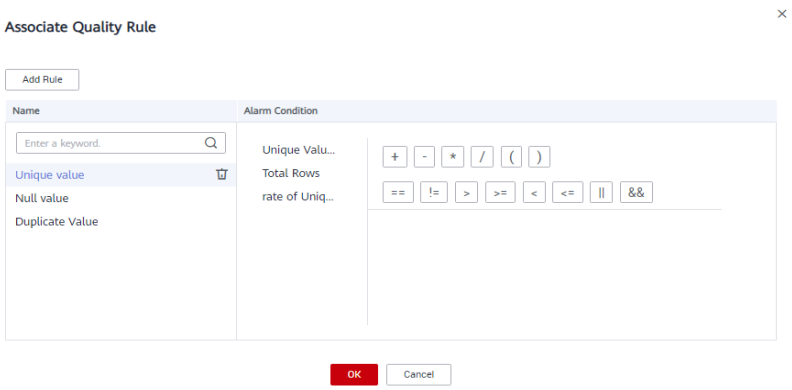
On the page for creating a data standard, only the selected parameters and custom parameters that have been added on the **Standard Templates** tab page of the **Configuration Center** are displayed. **Table 5-16** lists all parameters that are available in a data standard template. For details on how to configure a data standard template, see **Standard Templates**.

Table 5-16 Parameters for creating a data standard

Parameter	Description
*Standard Name	The value can contain only letters, digits, brackets, commas (,), spaces, and the following special characters: +#_[]/. It must start with a letter. If **Data Standard Allows Duplicate Names** is disabled, ensure that the standard name is unique in the current workspace. To check whether **Data Standard Allows Duplicate Names** is enabled, go to **DataArts Architecture** > **Configuration Center** > **Functions**.

Parameter	Description
*Standard Code	The value can be **Auto Generate** or **Custom**. The value must be unique in the current workspace. It is used to identify a data standard record. For details, see **Table 5-2**.
*Data Type	The possible values are **STRING**, **BIGINT**, **DOUBLE**, **TIMESTAMP**, **DATE**, **BOOLEAN**, and **DECIMAL**. The data type varies according to the system. The system converts the data type internally. If the required data type does not exist, you can add one. See **Field Types**.
Name (EN)	English name of the data standard It must start with a letter. Only letters, digits, brackets, spaces, and underscores (_) are allowed.
Data Length	Data length • You can leave this parameter blank. If it is left blank, there is no limit to the data length. • Select = and enter a number ranging from 1 to 10000. • Select ≤ and set a range from 1 to 10000. If you set this parameter and select **STRING** for **Data Type**, a data quality job will be created for the attribute matching the data standard. If you select any other data type, no data quality job will be created.
Allowed Value Exist	If **Allowed Value Exist** is enabled, you can specify one or more allowed values.
Allowed Value	This parameter is available only when **Allowed Value Exist** is enabled. You can type a value and press **Enter** to add it. You can add up to 20 allowed values.

Parameter	Description
Lookup Table	Select a created lookup table and the corresponding table fields. In this way, the lookup table fields can be associated with data standard. If no lookup table is created, create one. See **Creating Lookup Tables**. If **Create Data Quality Jobs** is selected for **Model Design Process** on the **Function Settings** tab page of **Configuration Center**, and the data standard of the referenced lookup tables is associated with the business tables in ER modeling, the system will automatically create quality jobs in DataArts Quality when the business tables are published, and generate quality rules based on the associated data standard and lookup tables. If the quality jobs have already been published, the system will automatically update the quality jobs and add the quality rules generated based on the data standard and lookup tables. If a public workspace is available, you need to manually set the reference lookup table source to **Public workspace** or **Current workspace** when selecting a lookup table in a common workspace. When **Public workspace** is enabled, lookup tables of the public workspace can be referenced in common workspace.

Parameter	Description
Quality Rule	This parameter is available if **Quality rule** is selected on the **Standard Templates** tab page on the **Configuration Center** page. You can associate a system quality rule or a quality rule you have created. Click  . In the dialog box displayed, click **Add Rule**. For example, add a rule named **Unique value**, select the rule, click **OK**, enter an alarm condition expression in the **Alarm Condition** text box, add other rules in the same way, and click **OK**. An alarm condition expression consists of alarm parameters and logical operators. When a quality job is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered. The alarm parameters of each data quality rule are listed as buttons. **Figure 5-54** Associate Quality Rule dialog box 
Rule Designer	Select a rule designer from the drop-down list box. This owner is responsible for making quality rules. You can enter an owner name or select an existing owner.
Rule Implementer	Select a rule implementer from the drop-down list box. This owner is responsible for implementing quality rules. You can enter an owner name or select an existing owner.
Level	• **global** indicates the global level. • **domain** indicates non-global level.
Custom Item	A custom item added on the **Standard Templates** tab page in **Metrics** > **Configuration Center**. You can add one or more custom items based on project requirements. For more information about adding custom items, see **Standard Templates**.

Parameter	Description
Description	A description of the data standard to create. Up to 600 characters are supported.

Step 3 Click **Save**.

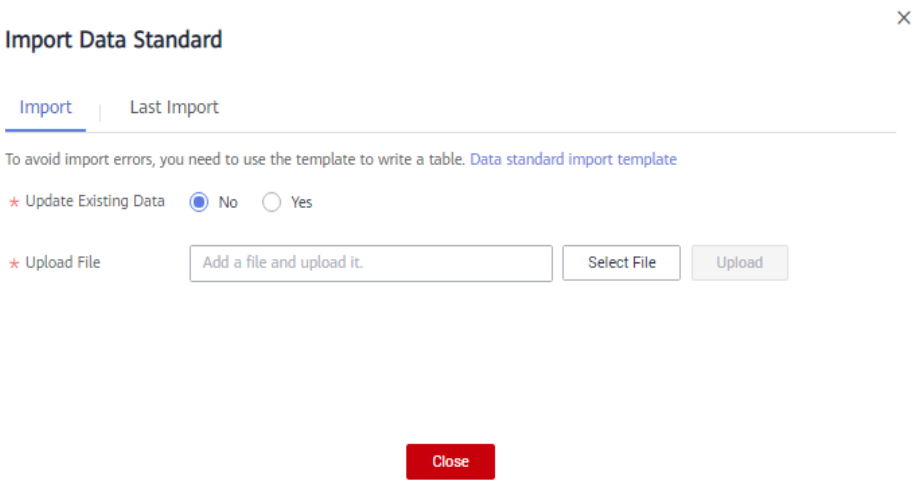
----End

Importing a Data Standard

Step 1 On the DataArts Architecture page, choose **Standards** > **Data Standards** in the left navigation pane.

Step 2 In the directory structure of data standards, select a directory and choose **More** > **Import**.

Figure 5-55 Import Data Standard dialog box



Step 3 In the **Import Data Standard** dialog box, determine whether to update the existing data. Existing data is uniquely identified by a standard code. If a standard code in the import template already exists in the current workspace, the system considers that the group of data to which the standard code in the import template belongs already exists.

Step 4 On the **Import** tab page, click **Data standard import template** to download the template. Open the template, set the parameters in the template based on service requirements, and save the settings.

Table 5-17 and Table 5-18 describe the parameters required for importing a data standard. Parameters whose names start with an asterisk (*) are mandatory, and other parameters are optional.

Table 5-17 Parameters in the Standards sheet

Parameter	Description
*Directory	The directory that the imported data standard belongs to.

Parameter	Description
*Standard Name	The name of the data standard to import. The value can contain only letters, digits, brackets, commas (,), spaces, and the following special characters: +#_[]/. It must start with a letter.
*Standard Code	You can select **Auto Generate** or **Custom**. The value must be unique in the workspace. It is used to identify a data standard record. For details, see **Table 5-2**.
*Data Type	The possible values are **STRING**, **BIGINT**, **DOUBLE**, **TIMESTAMP**, **DATE**, **BOOLEAN**, and **DECIMAL**. The data type varies according to the system. The system converts the data type internally. If the required data type does not exist, you can add one. See **Field Types**.
Data Length	Data length • You can leave this parameter blank. If it is left blank, there is no limit to the data length. • Enter a number ranging from 1 to 10000. • Set a range from 1 to 10000, for example **(1,20)**. If you enter a value and select **STRING** for **Data Type**, a data quality job will be created for the attribute matching the data standard. If you select any other data type, no data quality job will be created.
Allowed Value	The value **true** indicates that there are allowed values, and the value **false** indicates that there are no allowed values.
Allowed Value List	If you select **true** for **Allowed Value**, you must enter an allowed value. You can add up to 20 values. Multiple values must be separated by commas (,), for example, **1,2,3**.
Lookup Table	Set this parameter to the name of a created lookup table.
Lookup Table Field	If **Lookup Table** is not left blank, you must set **Lookup Table Field**. In this way, the code table field can be associated with the data standard.
Owner of Business Rules	Enter the business rule owner. You can enter the name of an owner or select an existing owner.
Owner of Data Monitoring	Enter the data monitoring owner. You can enter the name of an owner or select an existing owner.
Standard Level	• **global** indicates the global level. • **domain** indicates non-global level.

Parameter	Description
Description	A description of the data standard to import. Up to 600 characters are supported.
(Optional) Custom Item	If you have added one or more custom fields when customizing a data standard template, you must also fill in the corresponding fields in the import template. If no custom field is added, you do not need to fill in the fields. For details on how to customize a data standard template, see **Standard Templates**.

If **Quality rule** is selected on the **Standard Templates** tab page on the **Configuration Center** page, the downloaded template contains the **Quality Rules** sheet on which you can add quality rules for the data standard.

Table 5-18 Parameters in the Quality Rules sheet

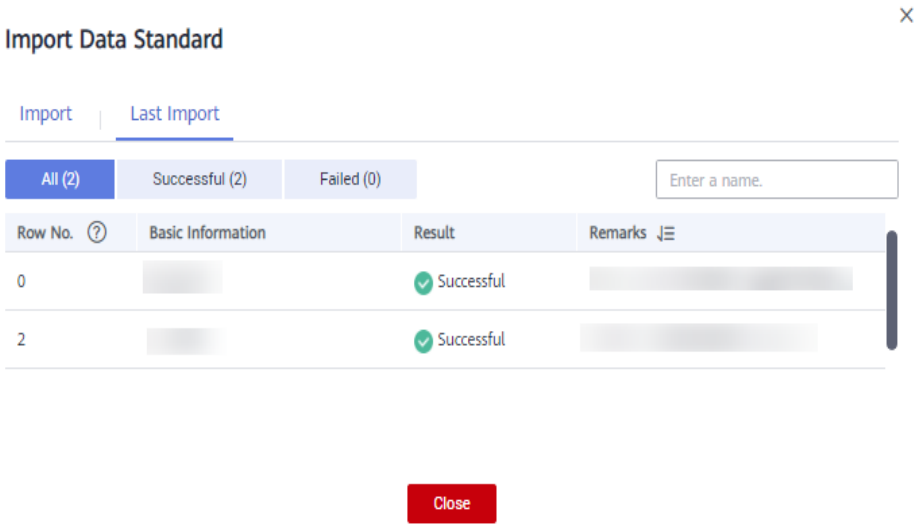
Parameter	Description
*Code	The code of the data standard that a quality rule is added to.
Rule Name	Enter an existing rule name. In the upper left corner of the DataArts Studio console, select **DataArts Quality** from the drop-down list box. Then, you can view the existing rule names on the **Rule Templates** page.
Alarm Config	An alarm condition expression consists of alarm parameters and logical operators. When a quality job is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered. In the alarm condition expression, alarm parameters are represented by variables such as *${1}*, *${2}*, and *${3}*. The variable name indicates the alarm parameter of the specified quality rule. The variable *$1* indicates the first alarm parameter, *$2* indicates the second alarm parameter, and so on. In the upper left corner of the DataArts Studio console, select **DataArts Quality** from the drop-down list box. Access the **Rule Templates** page and view the alarm parameters supported by the data quality rule in the **Result Description** column. Example: ${1} > 100
Expression	An expression must be configured when **Rule Name** is set to **Expression** or **Validity Verification**.

Step 5 Return to the **Import Data Standard** dialog box, select the data standard template file configured in the previous step, and click **Upload**.

If the uploaded template file fails the verification, modify the file and upload it again.

Step 6 In the **Import Data Standard** dialog box, the import result is displayed on the **Last Import** tab page. If the import is successful, click **Close**. If the import fails, you can view the failure cause, correct the template file, and upload it again.

Figure 5-56 Last Import tab page



----End

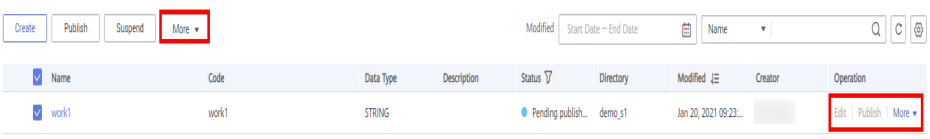
Managing a Data Standard

On the DataArts Architecture page, choose **Standards** > **Data Standards** in the left navigation pane. On the page displayed, you can manage data standards as required.

NOTE

- The data standards created in the **public workspace** can be queried in a common workspace, but the data standards created in a common workspace cannot be queried in the **public workspace**.
- A common workspace has the edit permission of only the data standards and directories created in the same workspace, and can view indexes in the **public workspace** rather than perform any operation on the data standards and directories in the **public workspace**.

Figure 5-57 List of data standards



On the **Data Standards** page, you can perform the following operations:

- **Search**

  Above the data standard list, select a filter such as the standard name, data type, creator, and reviewer, and click the search icon to search for data standards.

After locating the specified data standards, you can perform the following operations:

- Edit
- Publish
- Suspend

- **Import**

  Choose **More** > **Import** to import a data standard. Download the template, fill in it and upload it, and click **Close**.

- **Export**
  - Export data standards from a specified directory.

    In the data standard directory structure, select a directory and choose **More** > **Export** above the data standard list to export all data standards in the directory.
  - Export specified data standards.

    In the data standard list, select the data standards you want to export and choose **More** > **Export** above the list to export the selected data standards.

- **Delete**

  Select a data standard, and choose **More** > **Delete**. A data standard in publishing review, published, or suspension review state cannot be deleted. Referenced data standards cannot be deleted as well.

- **Publish**

  Select a data standard and click **Publish**. In the displayed dialog box, perform either of the following operations:

  - Select a reviewer. If no reviewer is available in the drop-down list, click + to add one.
  - Select **Auto-review**.

    NOTE

    Auto-review is available only when the current account is in the reviewer list.

  Click **OK**. If a reviewer is selected, the data standard is published after the application is approved. If **Auto-review** is selected, the data standard will be published immediately.

Exporting a Data Standard

Step 1 On the DataArts Architecture page, choose **Standards** > **Data Standards** in the left navigation pane.

Step 2 In the data standard directory structure, right-click a directory name and choose **Export**.

----End

5.6 Model Design

5.6.1 ER Modeling

5.6.1.1 Designing Physical Models

A physical model is a physical description about the conversion of elements such as entities, attributes, attribute constraints, and relationships from a logical model to a table relationship diagram that can be identified by database software using certain rules and methods.

On the **ER Modeling** page, you can create an SDI and a DWI layer. The models are implemented through physical modeling. In addition to converting a logical model to a physical model, you can directly create a physical model.

The following parts are included in this topic:

- **Considerations in Physical Model Design**
- **Creating a Physical Model**
- **Creating and Publishing a Table**
- **Importing a Physical Table by Reversing a Database**

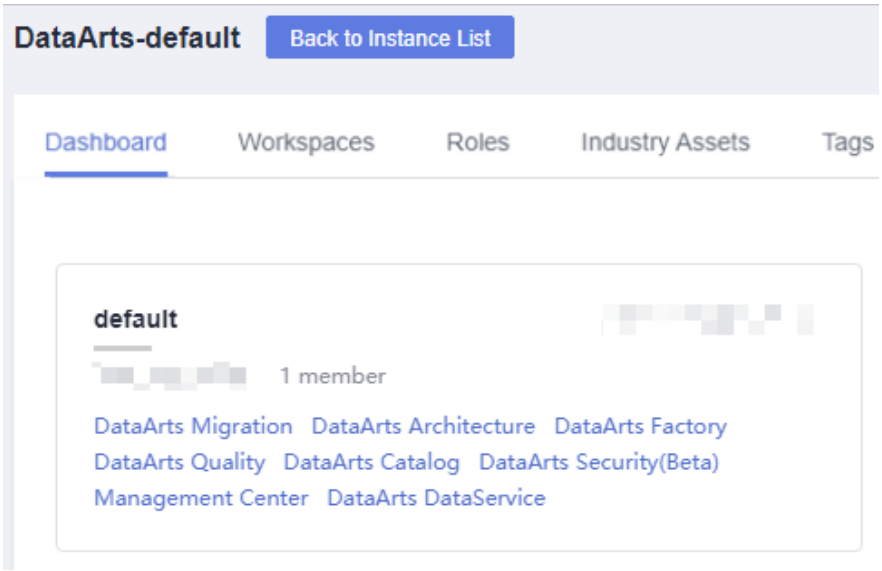
Considerations in Physical Model Design

- Physical models must ensure that the required functions are available and their performance is as good as expected.
- Physical models must ensure data consistency and quality.
- Few or no changes are made to the physical models when new services or functions are added.

Creating a Physical Model

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

Figure 5-58 DataArts Architecture



2. On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.
3. If no ER model is available, the **Create Hierarchical Governance Model** dialog box is displayed, prompting you to create models for the SDI and DWI layers. If there are physical models, click + in the **Physical Models** area to create a physical model.

   NOTE

   According to the data governance methodology and the ER and dimensional modeling methods, four layers of data warehouse models are available by default:

   - ER modeling consists of the SDI and DWI layers. Physical models belong to one of the two layers.
     - **SDI** stands for Source Data Integration and is the source data layer. SDI is a simple implementation of source system data.
     - **DWI** stands for Data Warehouse Integration, also called the data consolidation layer. DWI integrates and cleans data from multiple source systems, and implements entity relationship modeling based on the three normal forms.
   - In dimensional modeling, DWR-layer models are created based on dimensions, and data is aggregated into DM-layer models.
     - **Data Warehouse Report (DWR)** is based on the multi-dimensional model and its data granularity is the same as that of the DWI layer.
     - **Data Mart (DM)** is where multiple types of data are summarized. DM is designed to display the summarized data.

   The administrator can rename the models of the four layers by clicking next to the model names. The model name can contain only letters, digits, and underscores (_), and must start with a letter.

Figure 5-59 Creating a hierarchical governance model

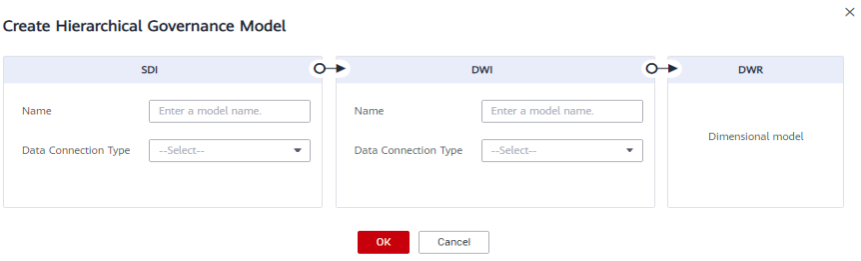
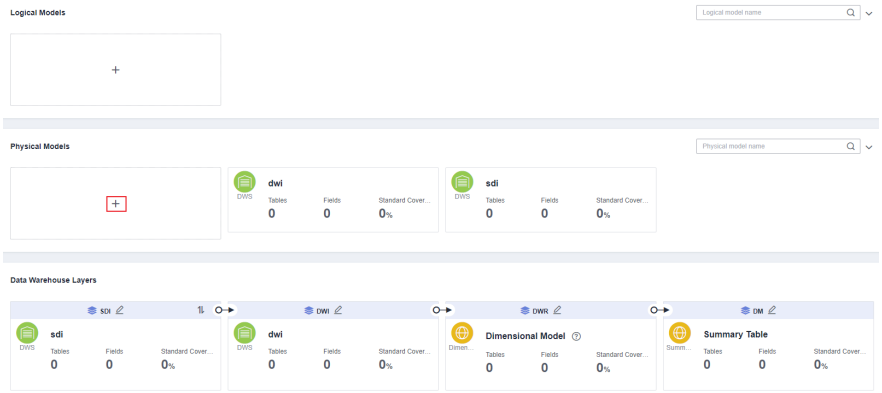


Figure 5-60 Creating a physical model



4. In the dialog box displayed, set the parameters and click **OK**.

Figure 5-61 Configuring the physical model

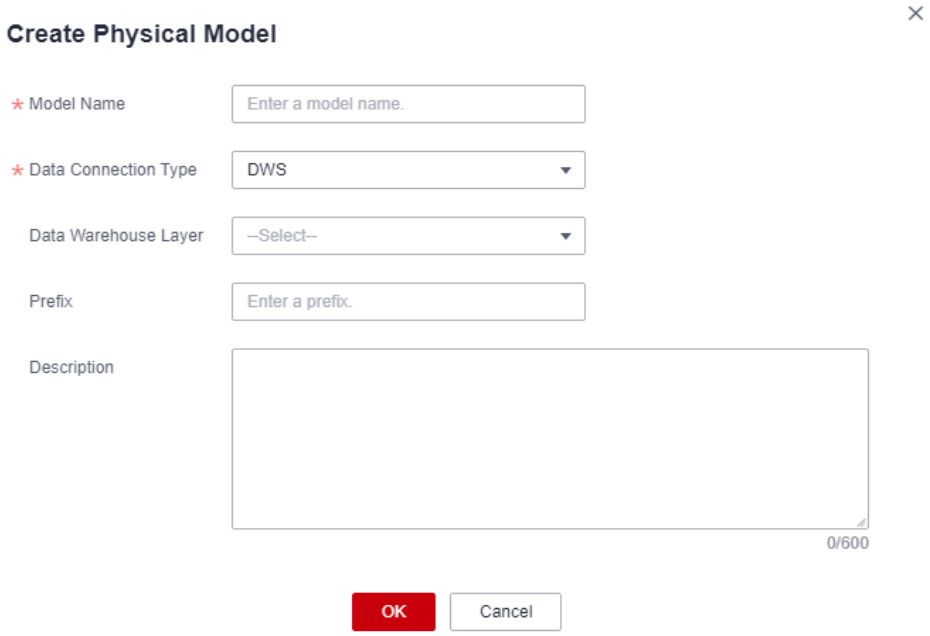


Table 5-19 Parameters for creating a physical model

Parameter	Description
*Name	Only letters, numbers, and underscores (_) are allowed.
*Data Connection Type	Select a data connection type from the drop-down list box.
Data Warehouse Layer	Select **SDI** or **DWI**. • **SDI** stands for Source Data Integration and is the source data layer. SDI is a simple implementation of source system data. • **DWI** stands for Data Warehouse Integration, also called the data consolidation layer. DWI integrates and cleans data from multiple source systems, and implements entity relationship modeling based on the three normal forms.
Prefix	Only letters, numbers, and underscores (_) are allowed. Dimension codes must start with letters. **NOTE** When you create, modify, or import a table to a physical model, the system checks whether a prefix is available. If no prefix is available, the verification fails. When you perform a reverse operation, the system also checks whether a prefix is available. If no prefix is available, the system automatically adds a prefix.

Parameter	Description
Description	A description of the ER model. Up to 600 characters are supported.

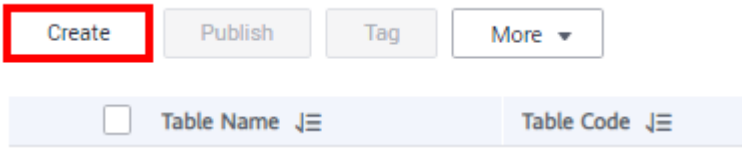
Creating and Publishing a Table

After creating an ER model, you can create a business table in the model.

Step 1 On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.

Step 2 Select the physical model for which you want to create a table, click the physical model to access the model management page, and click **Create**.

Figure 5-62 Entry for creating a table



Step 3 On the **Create Table** page, set the parameters as required.

1. Set the basic parameters.

Figure 5-63 Basic Settings tab page

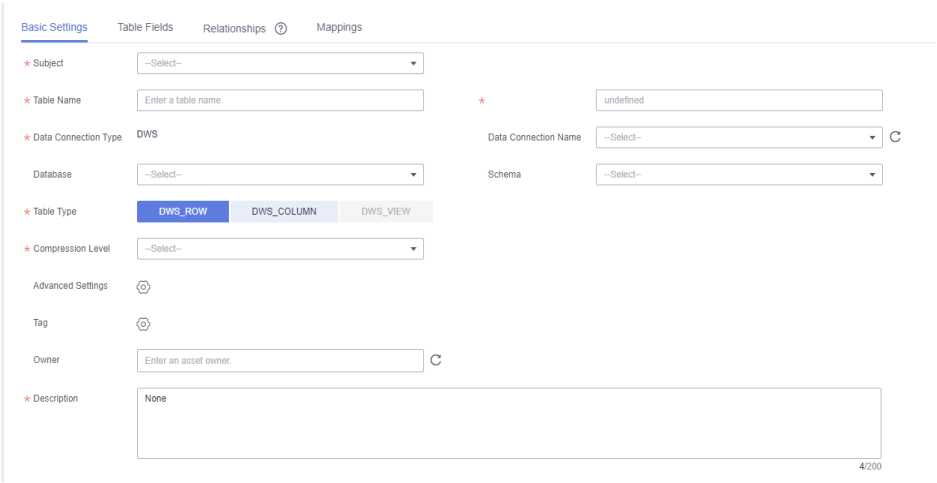


Table 5-20 Parameters on the Basic Settings tab page

Parameter	Description
*Subject	Select a subject from the drop-down list box.

Parameter	Description
*Table Name	The name of the table to create. The value can contain only letters, digits, brackets, commas (,), and the following special characters: +-#_[]/. It must start with a letter.
*Table Code	Name of the physical table converted from the logical entity. Table codes cannot start with numbers. Only letters, numbers, and the following special characters are allowed: _${}
*Data Connection Type	N/A
Data Connection	The name of the data connection. Select the required data connection. You are advised to use the same data connection for an ER model. If no data connection is available, access Management Center to create one. For details, see **Managing Data Connections**.
Database	The name of the database. Select a database from the drop-down list box.
Queue	DLI queue. This parameter is available only for DLI tables.
Schema	Schema of DWS or PostgreSQL This parameter is available only for DWS and PostgreSQL tables.
*Table Type	DLI models support the following table types: – **MANAGED**: Data is stored in a DLI table. – **EXTERNAL**: Data is stored in an OBS table. When **Table Type** is set to **EXTERNAL**, you must set **OBS Path**. The OBS path format is ***/bucket_name/filepath***. DWS models support the following table types: – **DWS_ROW**: Tables are stored to disk partitions by row. – **DWS_COLUMN**: Tables are stored to disk partitions by column. – **DWS_VIEW**: Tables are stored to disk partitions by view. The MRS Hive model supports **HIVE_TABLE** and **HIVE_EXTERNAL_TABLE**. The MRS Spark model supports **HUDI_COW** and **HUDI_MOR**. The PostgreSQL model supports only **POSTGRESQL_TABLE**. The MRS_CLICKHOUSE model supports only **CLICKHOUSE_TABLE**. The Oracle model supports only **ORACLE_TABLE**. The MySQL model supports only **MYSQL_TABLE**.

Parameter	Description
Path	This parameter is available only when the data source is MRS Hive and table type is HIVE_EXTERNAL_TABLE. The path can contain only letters, digits, slashes (/), periods (.), hyphens (-), underscores (_), and colons (:).
Compression Level	This parameter is available when the data connection type is DWS. The following compression levels are available for different table types: – **DWS_ROW**: **NO** and **YES** – **DWS_COLUMN**: **NO**, **LOW**, **MIDDLE**, and **HIGH**. – **DWS_VIEW**: The compression level is not supported.
Data Format	This parameter is available only for DLI tables. DLI models support the following table types: – **Parquet**: DLI can read non-compressed data or Parquet data that is compressed using Snappy and GZIP. – **CSV**: DLI can read non-compressed data or CSV data that is compressed using GZIP. – **ORC**: DLI can read non-compressed data or ORC data that is compressed using Snappy. – **JSON**: DLI can read non-compressed data or JSON data that is compressed using GZIP. – **Carbon**: DLI can read non-compressed Carbon data. – **Avro**: DLI can read non-compressed Avro data.
Advanced Settings	Set custom items to describe the table. The custom items can be viewed in the table details. For example, if you want to identify the source of the table, you can add item **source** and set its value to the table source information. Then you can view the table source information in the table details.
Tag	Tags are custom identifiers that help you classify and search for data assets. After adding a tag, you can search for related data assets in the DataArts Catalog module with ease. Click  . In the dialog box displayed, select one or more existing tags, or enter a new tag name and press **Enter**. Then press **OK**. You can also go to the **Tags** page of the DataArts Catalog module to add a tag. Then, return to this page and select the newly added tag from the drop-down list box. For details, see **Tags**. If you want to modify the tag of a table in ER modeling, you must suspend the table first. After modifying the tag, you can publish the table again.

Parameter	Description
Owner	You can enter an owner name or select an existing owner.
*Description	A description of the table. It allows 1 to 200 characters.
Associated Logical Entity	Select the logical entity to be associated with the table and the source model of the logical entity. You can also click the refresh button on the right. The system will automatically synchronize the source model with the same name as the physical table subject and the logical entity with the same name as the physical table. A logical entity can be associated with multiple physical tables.

2. Click **Add** to add required fields on the **Table Fields** page.

Figure 5-64 Adding required table fields

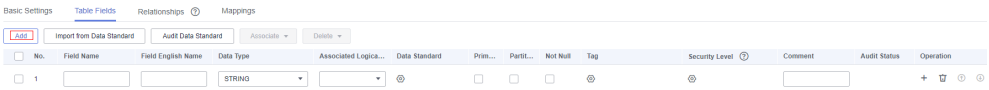


Table 5-21 Parameters on the Table Fields tab page

Parameter	Description
Name	The value can contain only letters, digits, brackets, commas (,), and the following special characters: +-#_[]/. It must start with a letter.
Code	Only letters, numbers, and underscores (_) are allowed. A field code must start with a letter.
Data Type	Field data type. If the required data type does not exist, you can add one. See **Field Types**.
Associated Logical Attribute	If the table configuration has been associated with a logical entity, you can select a logical attribute from the drop-down list box to associate it with the table field.

Parameter	Description
Data Standard	If you have created data standards, click to select one to associate with the field. If **Create Data Quality Jobs** is selected for **Model Design Process** on the **Function Settings** tab page in Configuration Center and a field is associated with a data standard, a quality job is automatically generated after a table is published. A quality rule is generated for each field associated with the data standard. The quality of the field is monitored based on the data standard. You can access the **Quality Job** page of DataArts Quality to view the job details. If no data standard is available, create one. See **Creating Data Standards** for details.
Primary Key	If this parameter is selected, the field is a primary key. **NOTE** If an MRS Spark connection is used to connect to MRS Hudi data sources, data can be written to the database only if fields have primary keys. Otherwise, table synchronization fails.
Partition	If this parameter is selected, the field is a partition field.
Not Null	Whether the parameter value can be left empty.
Tag	Click to add a tag. – In the dialog box displayed, select one or more existing tags. If no tag has been added, you can go to the **Tags** page of the DataArts Catalog module to add a tag. For details, see **Tags**. – In the dialog box displayed, enter a new tag name and press **Enter**. Tag names can contain letters, numbers, and underscores (_), but cannot start with underscores (_).
Security Level	You can click to add a security level for the logical entity attribute. If you cannot find the security level you want, click **go to** to go to the DataArts Security console and create a security level. You can disable this function on the **Models** tab page on the **Configuration Center** page.
Description	A description of the field to add.

3. (Optional) On the **Relationships** tab page, click **Add** to create a relationship.

   A relationship refers to the association between a parent and a child table (also called a primary and a secondary table). It describes how a table is associated with another table, or the impact of a table's behavior on another table. Relationships between tables in a data model are particularly important and must be accurately defined. Otherwise, the actual business rules cannot

be accurately described in the data model, and data consistency is greatly damaged.

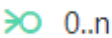
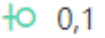
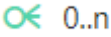
For example, if the **student ID** attribute of a score table is the primary key for a student table, the relationship between the two tables designed according to the third normal form (3NF) is as follows:

- Child table: score table
- Child table field FK: student ID
- Child to parent: # 1
- Parent table: student table
- Parent table field PK: student ID
- Parent to child: # 1

Figure 5-65 (Optional) Adding a relationship

Table 5-22 Parameters on the Relations tab page

Parameter	Description
Name	Name of the relationship
Child Table	Select a table from the drop-down list box. Click to set the current table as a child table. For example, if the **student ID** attribute of a score table is the primary key for a student table, the child table is the score table, and the corresponding parent table is the student table.
Child Table Field FK	Foreign key of the child table. The field of the child table must be the foreign key of the parent table. For example, if the **student ID** attribute of a score table is the primary key for a student table, the child table field FK is the student ID in the score table.

Parameter	Description
Child to Table	 indicates that each piece of data in the child table corresponds to only one piece of data in the parent table.  indicates that each piece of data in the child table corresponds to at most one piece of data in the parent table.  indicates that one piece of data in the child table corresponds to multiple pieces of data in the parent table.  indicates that each piece of data in the child table corresponds to at least one piece of data in the parent table.
Parent to Child	 indicates that each piece of data in the parent table corresponds to only one piece of data in the child table.  indicates that each piece of data in the parent table corresponds to at most one piece of data in the child table.  indicates that one piece of data in the parent table corresponds to multiple pieces of data in the child table.  indicates that one piece of data in the parent table corresponds to at least one piece of data in the child table.
Parent Table	Select the parent table corresponding to the selected child table. For example, if the **student ID** attribute of a score table is the primary key for a student table, the parent table is the student table, and the corresponding child table is the score table.
Parent Table Field PK	Primary key of the parent table. The field of the parent table must be the primary key of the parent table. For example, if the **student ID** attribute of a score table is the primary key for a student table, the parent table field PK is the student ID in the student table.
Role	You can customize a role name to identify the relationship.
Operation	Click  to delete a relationship. Click  to edit the relationship.

4. (Optional) On the **Mappings** tab page, click **Create** to create a mapping and design a data source based on the created mapping.
   - If the table field comes from different relationship models, you must create multiple mappings.

     Currently, table data can be obtained from ER models of different connection types. In each mapping, you only need to set the source field for the field that comes from the current mapping. Other fields do not need to be set.

     For example, if the data of the first five fields and the last five fields in the current table comes from two different models, create the following mappings:

     - **map1**: Create a table named **table01** from ER model A. In the **Field Mapping** area, set the source fields of the first to fifth fields to the corresponding fields with the same meaning in **table01**. The last five fields do not need to be set.
     - **map2**: Create a table named **table02** from ER model B. In the **Field Mapping** area, set the source fields of the sixth to tenth fields to the corresponding fields with the same meaning in **table02**. The first five fields do not need to be set.
   - If the field data in a table comes from multiple tables in the same ER model, you can create a mapping.

     In the source table of the mapping, you can set JOIN conditions for multiple tables, and then set source fields for the fields in the table. The selected source fields must have the same meanings as the fields in the table.

     For example, all fields in the current table come from ER model **d1**, the first, second, and third fields come from the **vendor**, **payment_type**, and **rate** tables respectively, and other fields come from the **dwd_taxi_trip_data** table.

     You can create a mapping, as shown in **Figure 5-66**. Join the **dwd_taxi_trip_data** table with the **vendor**, **payment_type**, and **rate** tables, and set the source fields in sequence in the field mapping.

   For details on the parameters for creating a mapping, see **Table 5-23**.

Figure 5-66 Configuring a mapping

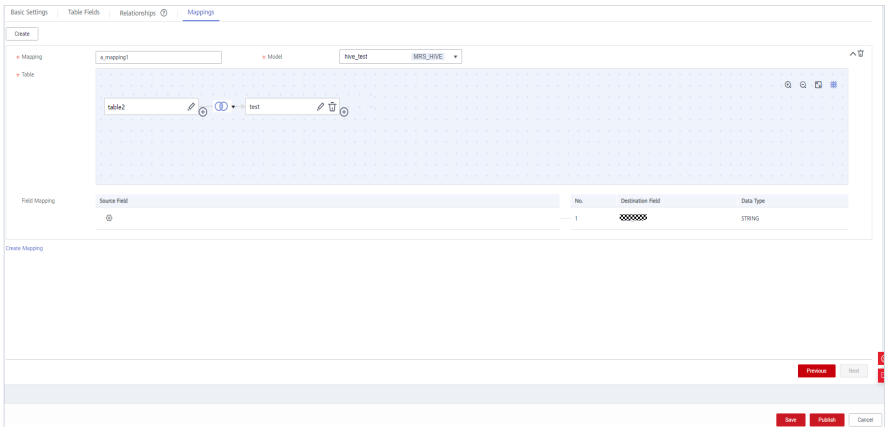
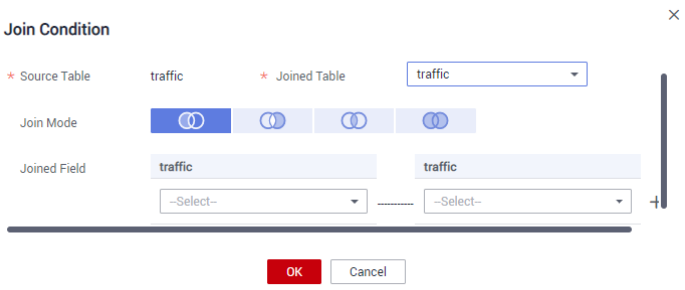


Table 5-23 Parameters of mappings

Parameter	Description
*Mapping	Only letters, numbers, and underscores (_) are allowed.
*Model	Select a created relationship model from the drop-down list box. If no relationship model has been created, create one. See **Designing Physical Models**.
*Table	Select a table from which data is obtained. If data is obtained from multiple tables, click next to the table name to set the JOIN condition between the table and other tables. 1. Select a JOIN mode. The JOIN mode includes left JOIN, right JOIN, inner JOIN, and outer JOIN from left to right. 2. Set the JOIN condition in the JOIN field. Generally, select the fields with the same meaning in the source table and joined table. Click + or − to add or delete a JOIN condition. The relationship between JOIN conditions is AND. 3. Click **OK**. 4. If you want to delete a joined table after setting the JOIN condition, click next to the table name. **Figure 5-67** Join Condition dialog box 
Field Mapping	Select a source field with the same meaning as the current mapping field. If a table field comes from multiple models, you must create multiple mappings. In each mapping, you only need to set the source field for the field that comes from the current mapping. Other fields do not need to be set.

In the upper right corner of the **Mappings** area, click to delete a mapping or click to collapse the mapping area.

5. (Optional) If the type of the new table is **DWS_VIEW**, click **Create** to create a view.

Figure 5-68 Creating a view

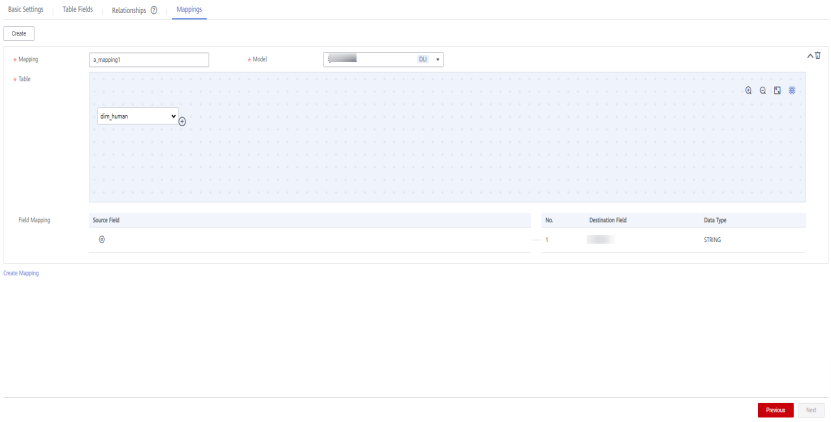
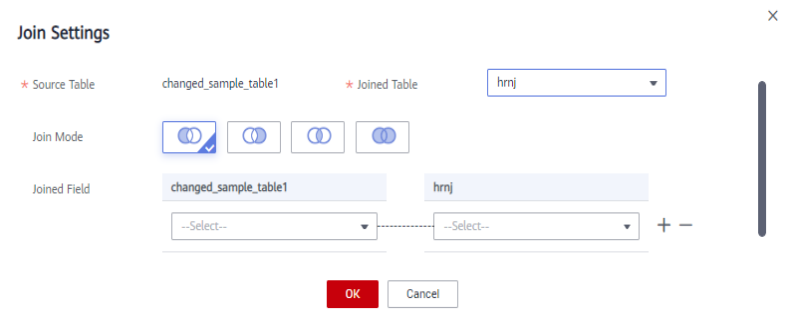


Table 5-24 Parameters

Parameter	Description
Mapping	Only letters, numbers, and underscores (_) are allowed.
Table	Select a table from which data is obtained. If data is obtained from multiple tables, click next to the table name to set the JOIN condition between the table and other tables. 1. Select a JOIN mode. The JOIN mode includes left JOIN, right JOIN, inner JOIN, and outer JOIN from left to right. 2. Set the JOIN condition in the JOIN field. Generally, select the fields with the same meaning in the source table and joined table. Click + or − to add or delete a JOIN condition. The relationship between JOIN conditions is AND. 3. Click **OK**. 4. If you want to delete a joined table after setting the JOIN condition, click next to the table name. **Figure 5-69** Join Settings dialog box 

Parameter	Description
Field Mapping	Select a source field with the same meaning as the current mapping field. If a table field comes from multiple models, you must create multiple mappings. In each mapping, you only need to set the source field for the field that comes from the current mapping. Other fields do not need to be set.

In the upper right corner of the **Mappings** area, click to delete a mapping or click to collapse the mapping area.

Step 4 Click **Publish**, select a reviewer, and click **Submit**.

NOTE

You can choose to publish the table to the production or development environment. By default, it is published to the production environment. If you do not choose an environment, the table cannot be published.

Step 5 Wait for the reviewer to approve the application. After the application is approved, return to the **ER Modeling** page to view the table status and synchronization status.

Publishing is an asynchronous operation. You can click to refresh the status. After table publishing application is approved, the system performs operations such as creating tables and synchronizing technical assets and business assets based on the configurations of **Model Design Process** on the **Function Settings** tab page in **Configuration Center**. The synchronization status is displayed in the **Sync Status** column of the table on the **Information Architecture** page.

- If the synchronization is successful, the table is successfully published. Move the cursor to in the **Sync Status** column. If the message indicating "creation succeeded" is displayed, the table has been successfully created in the corresponding data source.
- If one or more items fail to be synchronized, you can refresh the status. If the fault persists, choose **More** > **View History** and click the **Publish Log** tab to view logs.

  Troubleshoot the problem based on the logs. After the error is rectified, click **Resynchronize** on the **History** tab page to issue the synchronization command again. If the synchronization still fails, contact technical support for assistance.
- If **Synchronize logical assets** is enabled and **Physical Table Synchronize Logical Assets** is disabled, when you move the cursor to the icon for synchronizing logical assets in **Sync Status**, **Unsynchronized** is displayed.

NOTE

You can choose to synchronize the table to the production or development environment. By default, it is synchronized to the production environment. If you do not choose an environment, the table cannot be synchronized.

----End

Importing a Physical Table by Reversing a Database

By reversing databases, you can import one or more created database tables from other data sources into a physical table directory to turn them into physical tables.

Step 1 On the DataArts Architecture console, choose **ER Modeling** in the navigation pane on the left. Click a physical table to access it.

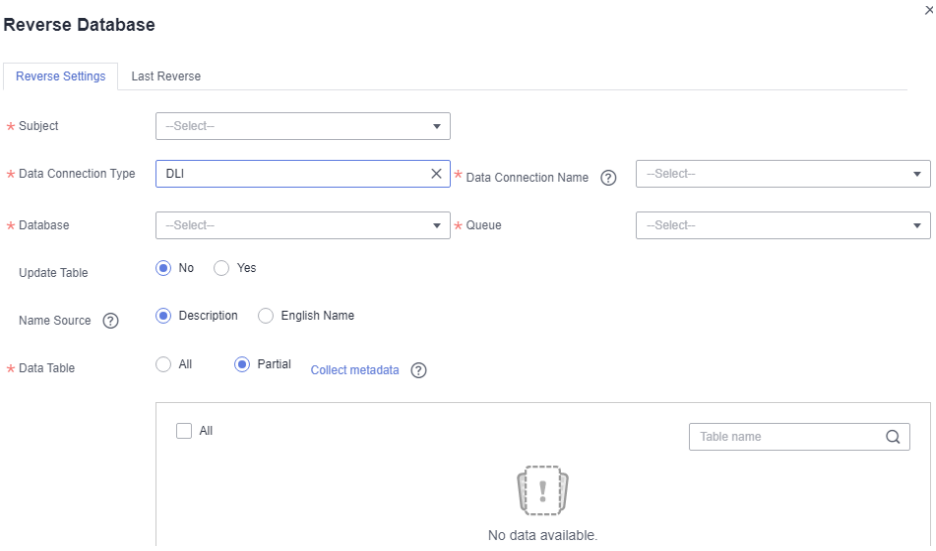
Step 2 Above the physical table list, click **Reverse Database**.

Step 3 In the displayed dialog box, set required parameters and click **OK**.

Table 5-25 Parameters for reversing the database

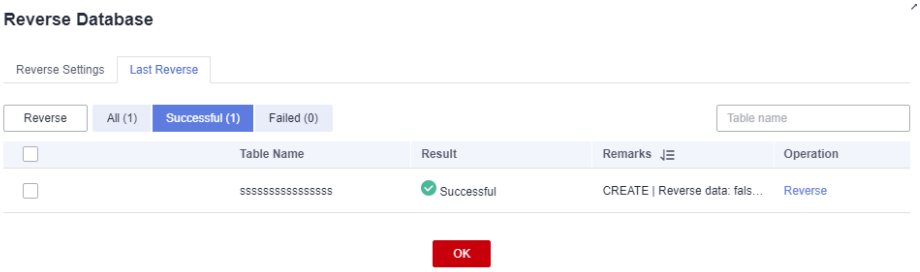
Parameter	Description
*Subject	Select a subject from the drop-down list.
*Data Connection Type	The data connection types supported by the reverse database are displayed in the drop-down list box. Select the required data connection type.
*Data Connection Name	Select a data connection. If you want to reverse a database from other data sources to a physical table directory, you must create a data connection in Management Center to connect to the data source. For details on how to create data connections, see **Managing Data Connections**.
*Database	Select a database.
*Schema	Select a value from the drop-down list box. This parameter is available only for DWS and PostgreSQL tables.
Queue	DLI queue. This parameter is available only when **Data Connection Type** is set to **DLI**.
Update Table	When **Yes** is selected, if the name of the reversed table is the same as that of an existing physical table, the existing physical table is updated.
*Data Table	You can select **All** or **Partial**.

Figure 5-70 Reverse Database dialog box



Step 4 You can view the result on the **Last Reverse** tab page. If the reverse operation is successful, click **Close**. If the reverse operation fails, you can view the failure cause. After the fault is rectified, select the table again and click **Reverse** to retry.

Figure 5-71 Last Reverse tab page



----End

5.6.1.2 Designing Logical Models

A logical model is an entity relationship diagram that accurately describes business rules based on entities and their relationships. Logical models must ensure the correctness and consistency of the data structure required by services and use a series of standard rules to reflect the features of various objects, and accurately define the relationships between entities.

In addition, logical models provide a reliable reference for constructing physical models and can be converted into physical models. Logical models are key to a successful database design.

The following parts are included in this topic:

- **Considerations in Logical Model Design**

- **Creating a Logical Model**
- **Creating and Publishing a Logical Entity**
- **Converting a Logical Model to a Physical Model**
- **Importing Logical Entities by Reversing a Database**

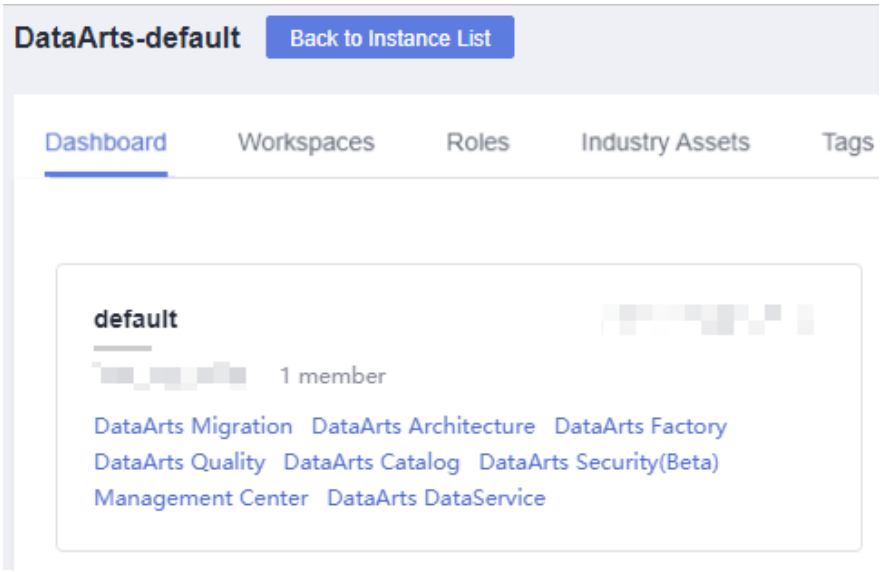
Considerations in Logical Model Design

- You must consider not only the current business status, but also the future business development.
- Personnel who are familiar with the businesses must participate in the modeling. In this way, the business requirements can be fully integrated into the models.
- Converting the logical model to the physical model must be efficient.
- You must consider physical features during physical modeling.
- Each entity, attribute, and relationship must be consistent with the information in the actual business.

Creating a Logical Model

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

   Figure 5-72 DataArts Architecture

2. On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.
3. If no ER model is available, the **Create Hierarchical Governance Model** dialog box is displayed, prompting you to create physical models for the SDI and DWI layers. After creating the physical models, you can click + in the **Logical Models** area to create logical models.

NOTE

According to the data governance methodology and the ER and dimensional modeling methods, four layers of data warehouse models are available by default:

- ER modeling consists of the SDI and DWI layers. Physical models belong to one of the two layers.
  - **SDI** stands for Source Data Integration and is the source data layer. SDI is a simple implementation of source system data.
  - **DWI** stands for Data Warehouse Integration, also called the data consolidation layer. DWI integrates and cleans data from multiple source systems, and implements entity relationship modeling based on the three normal forms.
- In dimensional modeling, DWR-layer models are created based on dimensions, and data is aggregated into DM-layer models.
  - **Data Warehouse Report (DWR)** is based on the multi-dimensional model and its data granularity is the same as that of the DWI layer.
  - **Data Mart (DM)** is where multiple types of data are summarized. DM is designed to display the summarized data.

The administrator can rename the models of the four layers by clicking next to the model names. The model name can contain only letters, digits, and underscores (_), and must start with a letter.

Figure 5-73 Creating hierarchical governance models

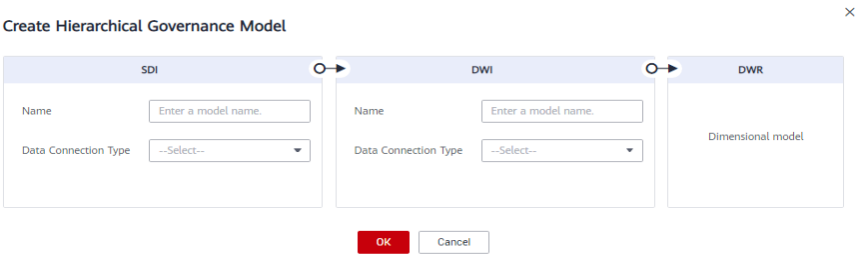
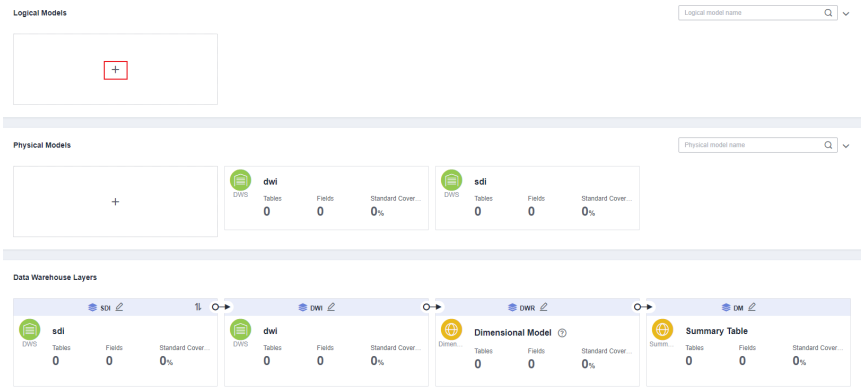


Figure 5-74 Creating a logical model



4. In the dialog box displayed, set the parameters and click **OK**.

Figure 5-75 Configuring the logical model

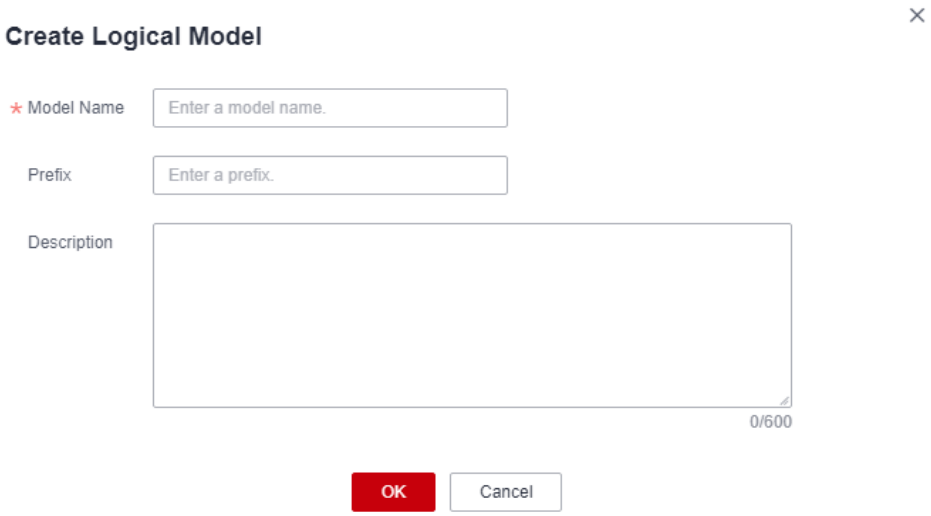


Table 5-26 Parameters for creating a logical model

Parameter	Description
*Name	Only letters, numbers, and underscores (_) are allowed.
Prefix	Only letters, numbers, and underscores (_) are allowed. Dimension codes must start with letters. **NOTE** When you create, modify, or import a logical entity to a logical model, the system checks whether a prefix is available. If no prefix is available, the verification fails. When you perform a reverse operation, the system checks whether a prefix is available. If no prefix is available, the system automatically adds a prefix.
Description	A description of the logical model.

Creating and Publishing a Logical Entity

A logical entity is a logical table. After creating a logical model, you can create a logical entity in the model.

Step 1 On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.

Step 2 On the displayed page, click a logical model to access its management page. Then, click **Create**.

Step 3 On the displayed page, configure parameters as prompted.

1. Set the basic parameters.

Figure 5-76 Basic Settings

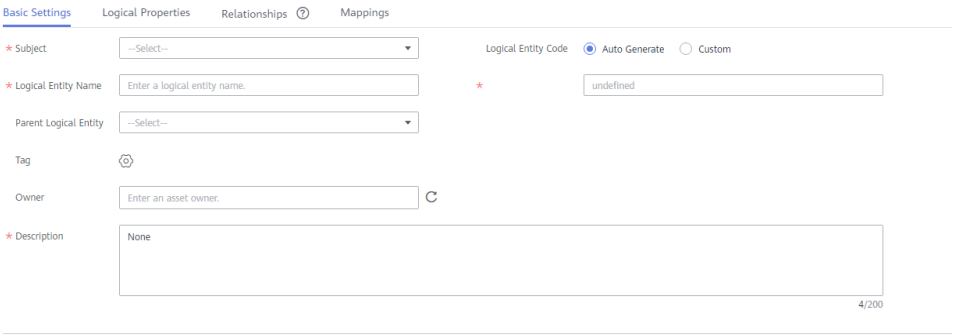


Table 5-27 Parameters on the Basic Settings tab page

Parameter	Description
* Subject	Select a subject from the drop-down list box.
Logical Entity Code	You can select **Auto Generate** or **Custom**.
* Table Name	Logic entity name. The value can contain only letters, digits, brackets, commas (,), and the following special characters: +-#_[]/. It must start with a letter.
* Table Code	Name of the physical table converted from the logical entity. Table codes cannot start with numbers. Only letters, numbers, and the following special characters are allowed: _${}
Parent Logical Entity	Set a parent logical entity, which is inherited by child logical entities. Common logical entities and attributes can be logically abstracted as a parent logical entity. After specific attributes are added to the parent logical entity, a child logical entity is generated. The modifications to the attributes in a parent logical entity affect all child logical entities that inherit it.
Tag	Tags are custom identifiers that help you classify and search for data assets. After adding a tag, you can search for related data assets in the DataArts Catalog module with ease. Click . In the dialog box displayed, select one or more existing tags, or enter a new tag name and press **Enter**. You can also go to the **Tags** page of the DataArts Catalog module to add a tag. Then, return to this page and select the newly added tag from the drop-down list box. For details, see **Tags**. If you want to modify the tag of a table in ER modeling, you must suspend the table first. After modifying the tag, you can publish the table again.
Owner	You can enter an owner name or select an existing owner.

Parameter	Description
* Description	A description of the table to create. It allows 1 to 200 characters.

2. On the **Logical Entity Attributes** page, add required attributes. **Table 5-28** lists the parameters for logical entity attributes.

Figure 5-77 Adding a logical entity attribute

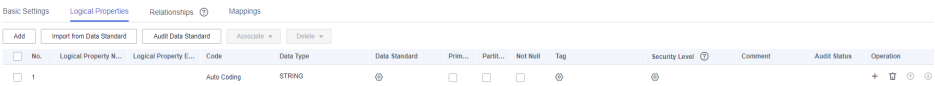


Table 5-28 Parameters for logical entity attributes

Parameter	Description
*Field Name	The value can contain only letters, digits, brackets, commas (,), and the following special characters: +-#_[]/. It must start with a letter.
*Field English Name	Only letters, numbers, and underscores (_) are allowed. A field code must start with a letter.
*Code	Code of the logical attribute. If the logical entity uses a custom code, the code of the logical attribute can be customized or automatically generated.
Data Type	Data type of the attribute. If you cannot find a desired data type from the drop-down list box, you can add a data type by referring to **Field Types**.

Parameter	Description
Data Standard	If you have created data standards, click  to select one to associate with the logical entity attribute. If **Create Data Quality Jobs** is selected for **Model Design Process** on the **Function Settings** tab page in Configuration Center and a logical entity attribute is associated with a data standard, a quality job is automatically generated after a logical entity attribute is published. A quality rule is generated for each logical entity attribute associated with the data standard. The quality of the logical entity attribute is monitored based on the data standard. You can access the **Quality Job** page of DataArts Quality to view the job details. If no data standard is available, create one. See **Creating Data Standards** for details. **NOTE** – After a logical entity is published, if the data standard code is modified, you must synchronize the dimension tables of the data standard to DataArts Catalog. Otherwise, the data standard code in the logical entity details cannot be updated.
Primary Key	If this parameter is selected, the attribute is a primary key. **NOTE** If you want to convert a logical model into a physical model, note the following restrictions for this parameter: If an MRS Spark connection is used to connect to MRS Hudi data sources, data can be written to the database only if fields have primary keys. Otherwise, table synchronization fails.
Partition	If this parameter is selected, the attribute is a partition field.
Not Null	Whether the parameter value can be left empty.
Tag	You can click  to add a tag for the logical entity attribute. – In the dialog box displayed, select one or more existing tags. If no tag has been added, you can go to the **Tags** page of the DataArts Catalog module to add a tag. For details, see **Tags**. – In the dialog box displayed, enter a new tag name and press **Enter**. Tag names can contain letters, numbers, and underscores (_), but cannot start with underscores (_).
Security Level	You can click  to add a security level for the logical entity attribute. If you cannot find the security level you want, click **go to** to go to the DataArts Security console and create a security level. You can disable this function on the **Models** tab page on the **Configuration Center** page.

Parameter	Description
Description	A description of the table to create.

3. On the **Relationships** tab page, click **Add** to create a relationship.

   A relationship refers to the association between a parent and a child entity (also called a primary and a secondary entity). It describes how an entity is associated with another entity, or the impact of an entity's behavior on another entity. Relationships between entities in a data model are particularly important and must be accurately defined. Otherwise, the actual business rules cannot be accurately described in the data model, and data consistency is greatly damaged.

   For example, if the **student ID** attribute of a score table is the primary key for a student table, the relationship between the two tables designed according to the third normal form (3NF) is as follows:

   - Child logical entity: score table
   - Child logical entity attribute FK: student ID
   - Child to parent: 1
   - Parent logical entity: student table
   - Parent logical entity attribute PK: student ID
   - Parent to child: 1

Figure 5-78 Adding a relationship

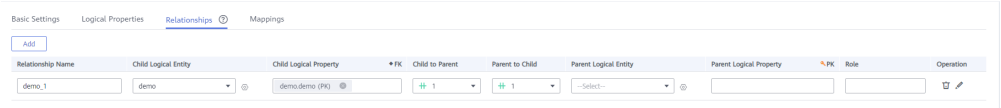


Table 5-29 Parameters on the Relationships tab page

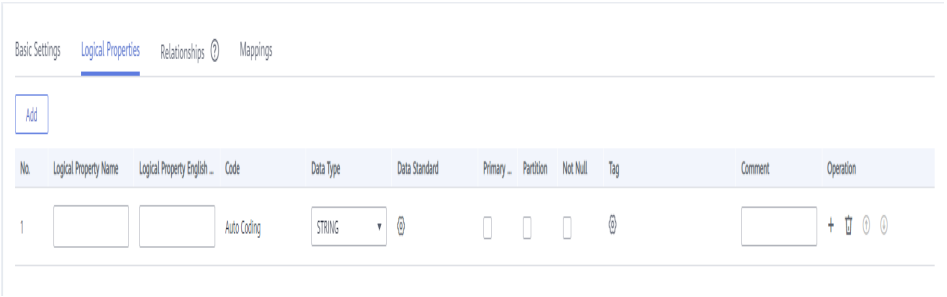
Parameter	Description
Name	Name of the relationship
Child Logical Entity	Select a child logical entity from the drop-down list box. Click to set the current logical entity as a child logical entity. For example, if the **student ID** attribute of a score table is the primary key for a student table, the child logical entity is the score table, and the corresponding parent logical entity is the student table.

Parameter	Description
Child Logical Entity Attribute FK	Foreign key of the child logical entity attribute. The attribute of the child logical entity must be the foreign key of the parent logical entity. For example, if the **student ID** attribute of a score table is the primary key for a student table, the foreign key of the child logical entity attribute is the student ID in the score table.
Child to Table	 indicates that each piece of data in the child logical entity corresponds to only one piece of data in the parent logical entity.  indicates that each piece of data in the child logical entity corresponds to at most one piece of data in the parent logical entity.  indicates that one piece of data in the child logical entity corresponds to multiple pieces of data in the parent logical entity.  indicates that one piece of data in the child logical entity corresponds to one piece of data in the parent logical entity at least.
Parent to Child	 indicates that the data in the parent logical entity is in one-to-one relationship with the data in the child logical entity.  indicates that each piece of data in the parent logical entity corresponds to at most one piece of data in the child logical entity.  indicates that one piece of data in the parent logical entity corresponds to multiple pieces of data in the child logical entity.  indicates that each piece of data in the parent logical entity corresponds to at least one piece of data in the child logical entity.
Parent Logical Entity	Select a logical entity that has a logical relationship with the selected child logical entity. For example, if the **student ID** attribute of a score table is the primary key for a student table, the parent logical entity is the student table, and the corresponding child logical entity is the score table.

Parameter	Description
Parent Logical Entity Attribute PK	Primary key of the parent logical entity attribute. The attribute of the parent logical entity must be the primary key of the parent logical entity. For example, if the **student ID** attribute of a score table is the primary key for a student table, the primary key of the parent logical entity attribute is the student ID in the student table.
Role	You can customize a role name to identify the relationship.
Operation	Click to delete a relationship. Click to edit the relationship.

4. On the **Mappings** page, click **Create** to create a mapping. Then click **Save**. Mapping means setting up a mapping relationship between the source and destination logical entity.

Figure 5-79 Creating a mapping



- **Mapping** is automatically generated when a mapping is created. You can change the value.
- **Source Logical Entity**: If data comes from multiple logical entities of a model, you can click next to a logical entity to establish a JOIN relationship between the logical entity and another logical entity.

Figure 5-80 Setting the JOIN condition for the source table

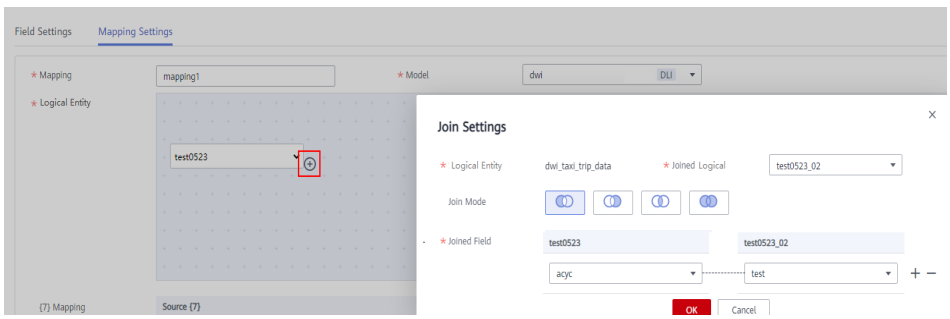


Table 5-30 JOIN conditions

Parameter	Description
Joined Logical Entity	Select a logical entity for which you want to establish a JOIN relationship with the source logical entity.
Joined Mode	Left JOIN, right JOIN, inner JOIN, and outer JOIN are represented from left to right.
Joined Attribute	Generally, the JOIN attribute in the source logical entity is the same as that in the joined logical entity. You can click + or − to add or delete a JOIN attribute. The relationship between JOIN attributes is AND.

- **Logical Attribute Mapping**: Select a source attribute with the same meaning as the current attribute.

Step 4 Click **Publish**, select a reviewer, and click **Submit**.

NOTE

You can choose to publish the logical model to the production or development environment. By default, it is published to the production environment. If you do not choose an environment, the logical model cannot be published.

Wait for the reviewer to approve the application. After the application is approved, return to the model list and view the created logical entity in the list.

NOTE

By default, **Synchronize logical assets** is selected for **Model Design Process** on the **Functions** tab page of the **Configuration Center** page.

- For new logical models, you can click **Publish** to synchronize them to the logical assets of the DataArts Catalog module.
- For historical logical models, you can click **More** and select **Synchronize** from the drop-down list box to synchronize them to the logical assets of the DataArts Catalog module.

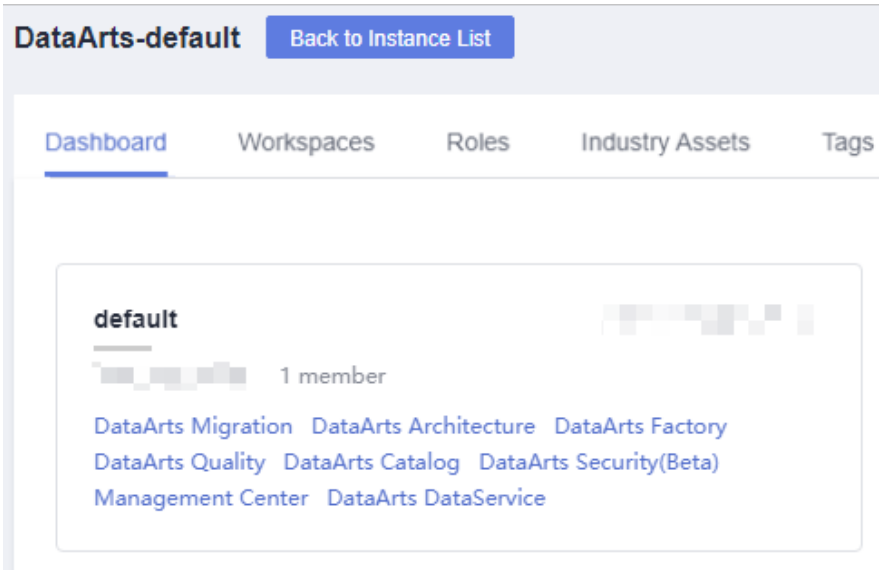
----End

Converting a Logical Model to a Physical Model

After a logical model is created, you can convert it to a new physical model or an existing physical model.

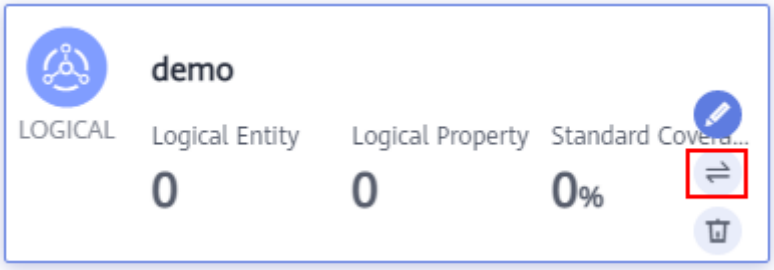
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

Figure 5-81 DataArts Architecture



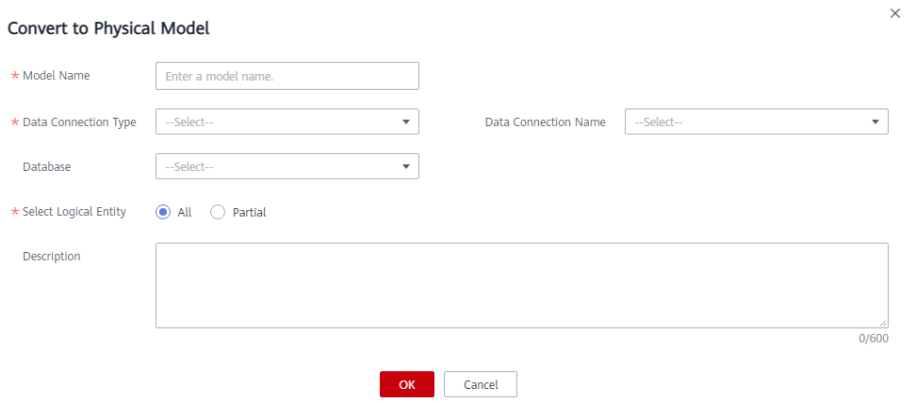
2. On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.
3. Find the required logical model and click the conversion button on the model.

Figure 5-82 Logical model conversion



4. In the **Convert to Physical Model** dialog box, set the parameters and click **OK**.

Figure 5-83 Convert to Physical Model dialog box



NOTE

During the conversion of a logical model into a physical model, the system checks whether the logical model has a prefix. If the model does not have a prefix, the system automatically adds a prefix to the model.

Table 5-31 Parameters

Parameter	Description
*Model Name	The name of the physical model to be converted from a logical model. You can enter a new model name, and then the system creates the model. You can also select an existing model name from the drop-down list box. Only letters, numbers, and underscores (_) are allowed.
*Update Existing Table	This parameter is displayed when a model name is selected. • No • Yes If you select **Yes**, you need to set **Physical Table Update Mode**. – Retain unnecessary fields – Delete unnecessary fields
*Data Connection Type	Select a data connection type from the drop-down list box.
Data Connection	The name of the data connection. Select the required data connection. You are advised to use the same data connection for an ER model. If no data connection is available, access Management Center to create one. For details, see **Managing Data Connections**.
Database	The name of the database. Select a database from the drop-down list box. If no database is available, access DataArts Factory to create one. For details, see **Creating a Database**.
Select Logical Entity	• **All**: Convert all logical entities into physical tables. • **Partial**: Convert the selected logical entities into physical tables.
Queue	DLI queue. This parameter is available only for DLI data connections.
Schema	Schema of DWS or POSTGRESQL. This parameter is available only for DWS and PostgreSQL data connections.
Description	A description of the model. Up to 600 characters are supported.

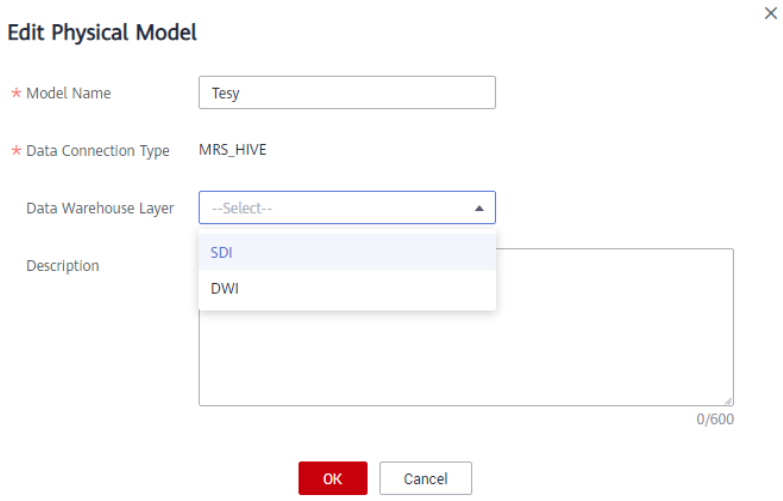
5. After the model is converted to a physical model, you can set layers for the physical model. You can select the SDI or DWI layer. As shown in **Figure 5-84**, move the cursor to the card of the physical model and click the edit button of the model.

Figure 5-84 Setting layers for the physical model

In the displayed dialog box, select **SDI** or **DWI** for **Data Warehouse Layer**.

- **SDI** stands for Source Data Integration and is the source data layer. SDI is a simple implementation of source system data.
- **DWI** stands for Data Warehouse Integration, also called the data consolidation layer. DWI integrates and cleans data from multiple source systems, and implements entity relationship modeling based on the three normal forms.

Figure 5-85 Editing the physical model



Importing Logical Entities by Reversing a Database

By reversing databases, you can import one or more created database tables from other data sources into a logical entity directory to turn them into logical entities.

Step 1 On the DataArts Architecture console, choose **ER Modeling** in the navigation pane on the left. Click a logical entity to access it.

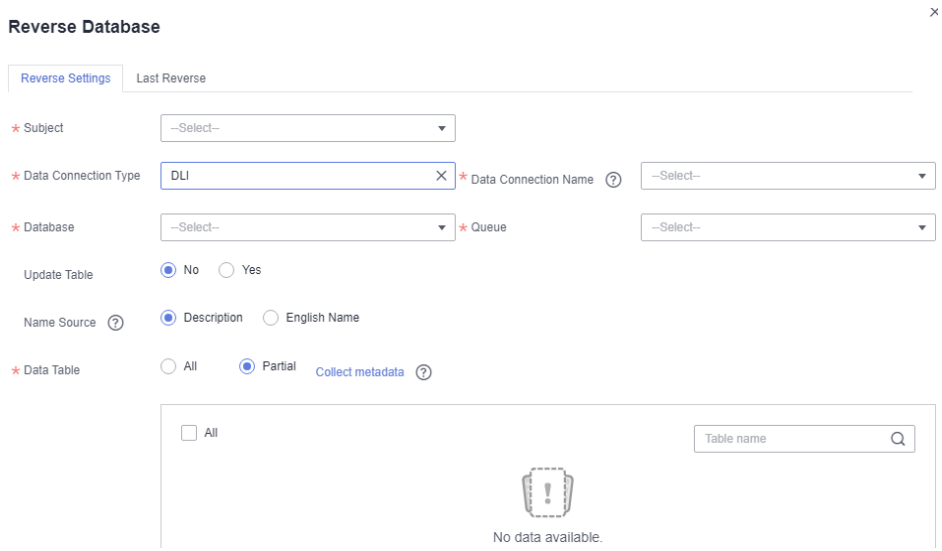
Step 2 Above the logical entity list, click **Reverse Database**.

Step 3 In the displayed dialog box, set required parameters and click **OK**.

Table 5-32 Parameters for reversing the database

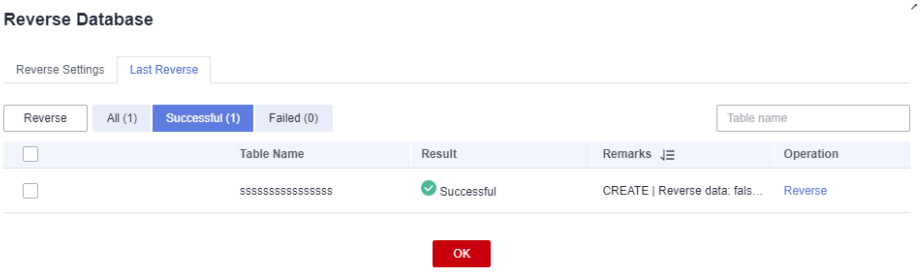
Parameter	Description
*Subject	Select a subject from the drop-down list.
*Data Connection Type	The data connection types supported by the reverse database are displayed in the drop-down list box. Select the required data connection type.
*Data Connection Name	Select a data connection. If you want to reverse a database from other data sources to a logical entity directory, you must create a data connection in Management Center to connect to the data source. For details on how to create data connections, see **Managing Data Connections**.
*Database	Select a database.
*Schema	Select a value from the drop-down list box. This parameter is available only for DWS and PostgreSQL tables.
Queue	DLI queue. This parameter is available only when **Data Connection Type** is set to **DLI**.
Update Table	When **Yes** is selected, if the name of the reversed table is the same as that of an existing table in the logical entity list, the logical entity is updated.
*Data Table	You can select **All** or **Partial**.

Figure 5-86 Reverse Database dialog box



Step 4 You can view the result on the **Last Reverse** tab page. If the reverse operation is successful, click **Close**. If the reverse operation fails, you can view the failure cause. After the fault is rectified, select the table again and click **Reverse** to retry.

Figure 5-87 Last Reverse tab page



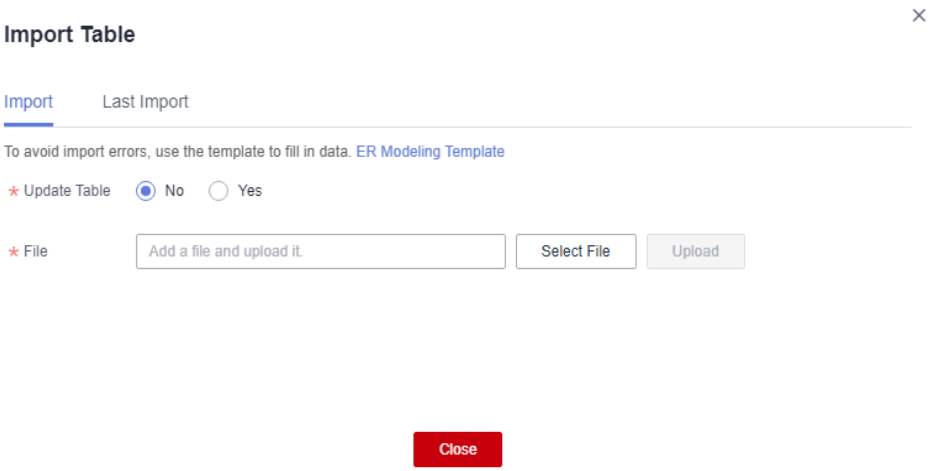
----End

Importing Logical Entities

Importing an Excel file

1. Above the logical entity list, click **Import** and select **Import EXCEL**. In the **Import Table** dialog box, click the **Import** tab and then **ER Modeling Template**.

   Figure 5-88 Importing an Excel file

2. Edit and save the downloaded template.
3. Choose whether to update existing data.

   NOTE

   If a code in the template already exists in the system, the data is considered duplicate.

   - **No**: If the data to be imported already exists in the system, the existing data in the system will not be replaced.
   - **Yes**: If the data to be imported already exists in the system:
     - If the existing data in the system is in draft state, the data will be replaced and new draft data will be generated.
     - If the existing data in the system is in published state, expanded data will be generated.

4. Click **Select File** and select the template you have edited and saved.
5. Click **Upload**. When the template is uploaded, the **Last Import** page is displayed. You can view the imported data.
6. Click **Close**.

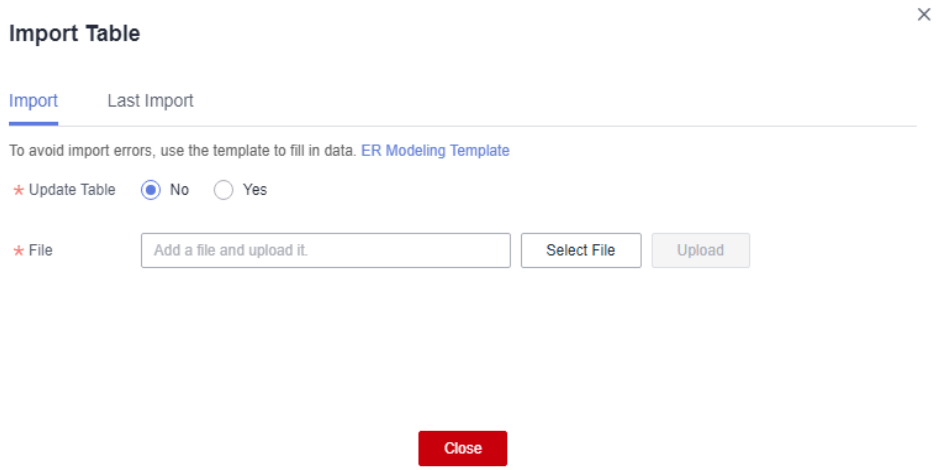
Importing an LDM model

NOTE

- Before importing an LDM model, select a theme. Otherwise, the model cannot be imported.
- Only logical models can be imported.
- Prepare an .ldm logical model exported from the third-party system Powet Designer in advance.
- LDM models of version 16.x can be imported.

1. Above the logical entity list, click **Import** and select **Import LDM**. In the **Import Table** dialog box, click the **Import** tab.

Figure 5-89 Importing an LDM model



2. Choose whether to update existing data.
   - **No**: If the data to be imported already exists in the system, the existing data in the system will not be replaced.
   - **Yes**: If the data to be imported already exists in the system:
     - If the existing data in the system is in draft state, the data will be replaced and new draft data will be generated.
     - If the existing data in the system is in published state, expanded data will be generated.
3. Click **Select File** and select the prepared .ldm logical model.
4. Click **Upload**. When the template is uploaded, the **Last Import** page is displayed. You can view the imported data.
5. Click **Close**.

Exporting Models

1. Above the logical entity list, click **Export**.
2. Select the object to be exported.

   Select **Table** or **DDL**.

   If you select **DDL**, select **ALL** or **Partial** for **Scope**. If you select **Partial**, select the tables to be exported.
3. Click **OK**.

5.6.2 Dimensional Modeling

5.6.2.1 Creating Dimensions

A dimension is the perspective to observe and analyze business data and assist in data aggregation, drilling, slicing, and analysis, and used as a GROUP BY condition in SQL statements. Most dimensions have hierarchical structures, such as geographic dimensions (including countries, regions, provinces/states, and cities) and time dimensions (including annually, quarterly, and monthly dimensions).

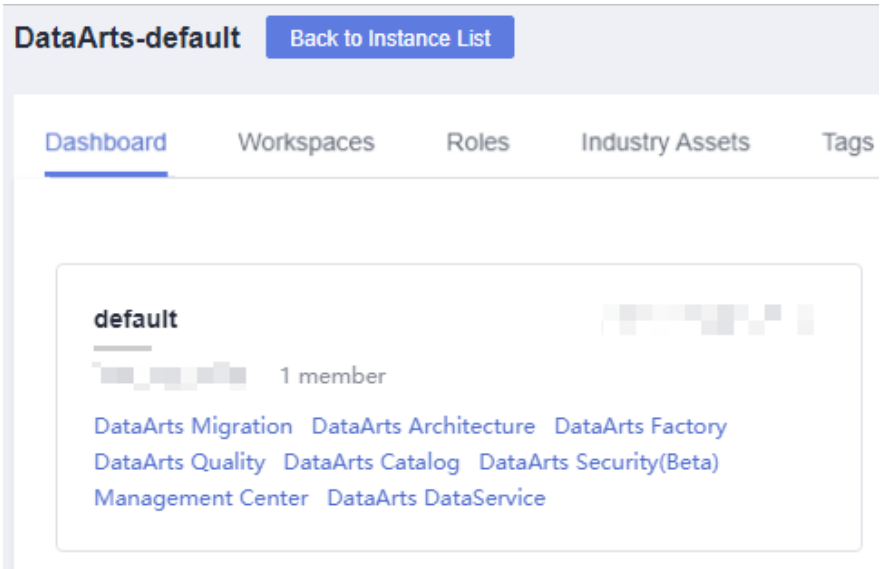
Impact on the System

After a dimension is published and approved, the system automatically creates a dimension table corresponding to the dimension. The name and code of the dimension table are the same as those of the dimension.

Creating and Publishing a Dimension

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

   Figure 5-90 DataArts Architecture

2. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane. On the displayed page, click the **Dimensions** tab.

3. Select an object from the subject directory on the left and click **Create**.

   Before creating a dimension, ensure that a subject is available. For details on how to add a subject, see **Designing Subjects**.

4. On the page displayed, set the parameters.

   Set the basic settings and physicalization settings as described below.

Figure 5-91 Dimension parameters

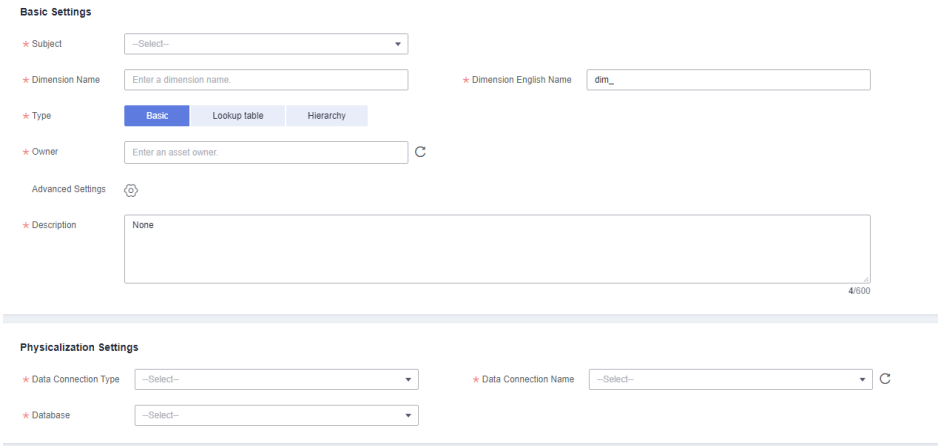


Table 5-33 Parameters in the Basic Settings area

Parameter	Description
*Subject	Select a subject from the drop-down list box.
*Dimension Name	It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_
*Dimension English Name	Only letters, numbers, and underscores (_) are allowed. Dimension codes must start with letters.
*Type	• **Basic**: a dimension that does not have a hierarchical structure. • **Lookup Table**: a dimension created based on a lookup table. The field information and data of the dimension are the same as those of the lookup table, indicating that the content is an enumerable dimension. • **Hierarchy**: a dimension with a hierarchical structure between attributes.
Advanced Settings	Set custom items to describe the table. The custom items can be viewed in the table details. For example, if you want to identify the source of the table, you can add item **source** and set its value to the table source information. Then you can view the table source information in the table details.

Parameter	Description
*Owner	You can enter an owner name or select an existing owner.
*Description	A description of the dimension to create. It allows 1 to 600 characters.

Table 5-34 Parameters in the Physicalization Settings area

Parameter	Description
*Data Connection Type	Select a data connection type from the drop-down list box.
*Data Connection Name	The name of the data connection. Select the required data connection. If no data connection is available, access Management Center to create one. For details, see **Managing Data Connections**.
*Database	The name of the database. Select a database from the drop-down list box. If no database is available, access DataArts Factory to create one. For details, see **Creating a Database**.
Queue	DLI queue. This parameter is displayed only for DLI data connections.
Schema	DWS or POSTGRESQL mode. This parameter is displayed only for DWS and POSTGRESQL data connections.

Parameter	Description
Table Type	DLI models support the following table types: • **MANAGED**: Data is stored in a DLI table. • **EXTERNAL**: Data is stored in an OBS table. When **Table Type** is set to **EXTERNAL**, you must set **OBS Path**. The OBS path format is ***/bucket_name/filepath***. DWS models support the following table types: • **DWS_ROW**: row-store table. Tables are stored to disk partitions by row. • **DWS_COLUMN**: column-store table. Tables are stored to disk partitions by column. • **DWS_VIEW**: view-store table. Tables are stored to disk partitions by view. The MRS Hive model supports **HIVE_TABLE** and **HIVE_EXTERNAL_TABLE**. The MRS Spark model supports **HUDI_COW** and **HUDI_MOR**. The PostgreSQL model supports only **POSTGRESQL_TABLE**. The MRS ClickHouse model supports only **CLICKHOUSE_TABLE**. The Oracle model supports only **ORACLE_TABLE**. The MySQL model supports only **MYSQL_TABLE**.
Compression Level	This parameter is available when the data connection type is DWS. The following compression levels are available for different table types: • **DWS_ROW**: **NO** and **YES** • **DWS_COLUMN**: **NO**, **LOW**, **MIDDLE**, and **HIGH**. • **DWS_VIEW**: The compression level is not supported.

Parameter	Description
Distributed By	This parameter is displayed only for DWS data connections. You can select multiple fields. • **REPLICATION**: A full table is stored on each DN. The advantage of this option is that each DN has all the data of a table. During the join operation, data redistribution can be avoided, reducing network overhead. The disadvantage is that each DN retains the complete data of a table, resulting in data redundancy. Generally, this option is recommended for small dimension tables. • **HASH**: If you select this option, you must specify a distribution key for the user table. When a record is inserted, the system performs hash computing based on values in the distribute keys and then stores data on the corresponding DN. In a Hash table, I/O resources on each node can be used during data read/write, which improves the read/write speed of a table. Generally, this option is recommended for large dimension tables (a large dimension table contains over 1 million records).
PreCombineField	This parameter is available only for Spark data connections.
Path	This parameter is available only when the data source is MRS Hive and table type is HIVE_EXTERNAL_TABLE. The path can contain only letters, digits, slashes (/), periods (.), hyphens (-), underscores (_), and colons (:).

Add dimension fields in the **Attribute Settings** area. You can click **Add** to add multiple dimension fields.

Figure 5-92 Field configuration

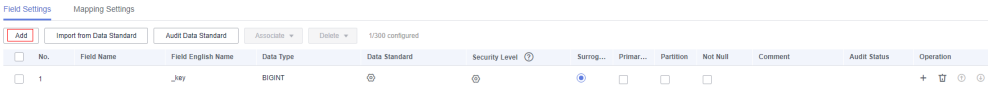


Table 5-35 Parameters in the Attribute Settings area

Parameter	Description
Name	It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_
Field Code	Field codes must start with letters. Only letters, numbers, and underscores (_) are allowed.
Data Type	Type of data defined based on the original data.

Parameter	Description
Data Standard	Click to select a data standard to be associated with the field. If **Create Data Quality Jobs** is selected for **Model Design Process** on the **Function Settings** tab page in Configuration Center and a field is associated with a data standard, a quality job is automatically generated after a dimension is published. A quality rule is generated for each field associated with the data standard. The quality of the field is monitored based on the data standard. You can access the **Quality Job** page of DataArts Quality to view the job details. If no data standard is available, create one. See **Creating Data Standards** for details.
Security Level	You can click to add a security level for the logical entity attribute. If you cannot find the security level you want, click **go to** to go to the DataArts Security console and create a security level. You can disable this function on the **Models** tab page on the **Configuration Center** page.
Surrogate Key	Select a field as the surrogate key based on project requirements. By default, the first dimension attribute is the surrogate key.
Primary Key	Select a field as the primary key based on project requirements. **NOTE** If an MRS Spark connection is used to connect to MRS Hudi data sources, data can be written to the database only if fields have primary keys. Otherwise, table synchronization fails.
Partition	Whether to be set as a partition field.
Not Null	Whether the parameter value can be left empty.
Description	A description of the dimension field you add.
Audit Status	Whether to audit the data standard
Operation	Related operations

On the **Mapping Settings** tab page, click **Create** to create a mapping between dimensions and physical tables. Set the parameters.

Figure 5-93 Mapping settings

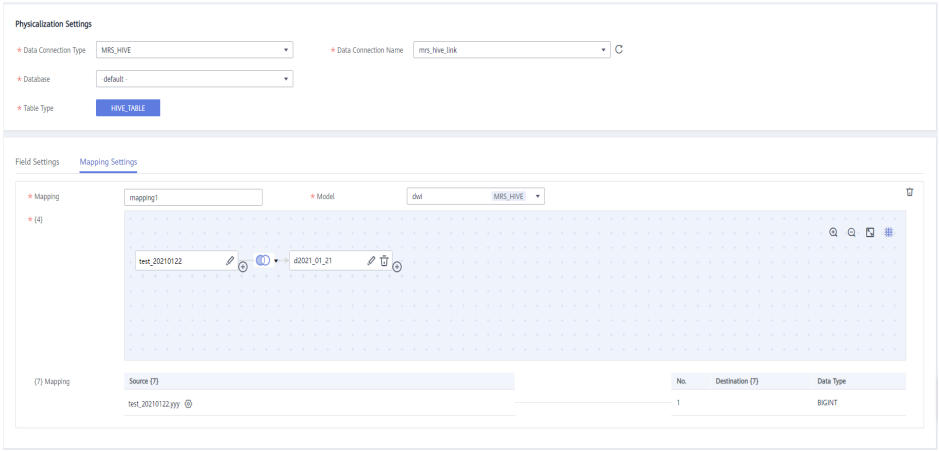
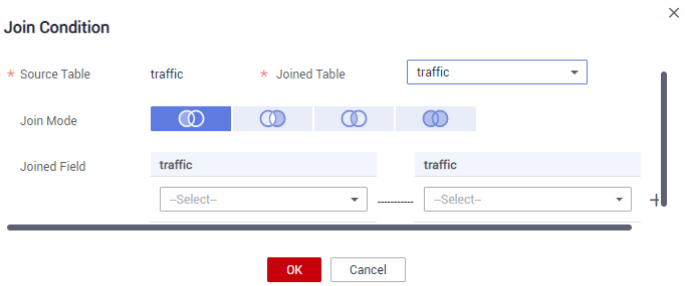


Table 5-36 Parameters of mappings

Parameter	Description
*Mapping	Only letters, numbers, and underscores (_) are allowed.
*Model	Select a created relationship model from the drop-down list box. If no relationship model has been created, create one. See **Designing Physical Models**.

Parameter	Description
*Table	Select a table from which data is obtained. If data is obtained from multiple tables, click next to the table name to set the JOIN condition between the table and other tables. 1. Select a JOIN mode. The JOIN mode includes left JOIN, right JOIN, inner JOIN, and outer JOIN from left to right. 2. Set the JOIN condition in the JOIN field. Generally, select the fields with the same meaning in the source table and joined table. Click + or − to add or delete a JOIN condition. The relationship between JOIN conditions is AND. 3. Click **OK**. 4. If you want to delete a joined table after setting the JOIN condition, click next to the table name. **Figure 5-94** Join Condition dialog box 
Field Mapping	Select a source field with the same meaning as the current mapping field. If a table field comes from multiple models, you must create multiple mappings. In each mapping, you only need to set the source field for the field that comes from the current mapping. Other fields do not need to be set.

In the upper right corner of the **Mappings** area, click to delete a mapping or click to collapse the mapping area.

5. Click **Publish**.

   NOTE

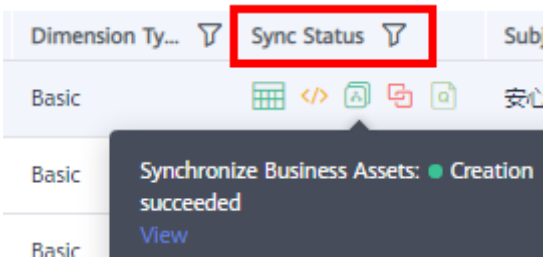
   You can choose to publish the dimension to the production or development environment. By default, it is published to the production environment. If you do not choose an environment, the dimension cannot be published.

6. In the dialog box displayed, select a reviewer and click **OK**.
7. Repeat **3** to **6** to create and publish other dimensions.
8. All dimensions must be approved by reviewers.

After the application is approved, the system automatically creates a dimension table corresponding to the dimension. The name and code of the dimension table are the same as those of the dimension. On the **Dimensional Modeling** page, click the **Dimension Tables** tab to view the created dimension table.

In the dimension table list, you can view the synchronization status of the dimension table in the **Sync Status** column.

Figure 5-95 Sync Status of the dimension table



- If the synchronization is successful, the dimension is successfully published and the dimension table is successfully created in the database.
- If the synchronization failed, click **View History** in the row where the dimension table is located. On the page displayed, click the **History** tab to view logs. Troubleshoot the problem based on the logs. After the error is rectified, go back to the dimension table list and click **Synchronize** above the dimension table list to issue the synchronization command again. If the problem persists, contact technical support personnel.

  NOTE

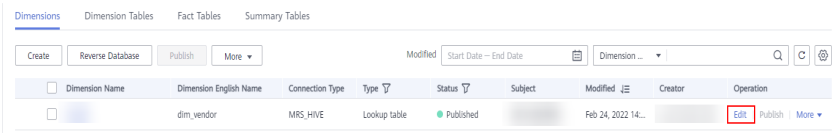
  You can choose to synchronize the table to the production or development environment. By default, it is synchronized to the production environment. If you do not choose an environment, the table cannot be synchronized.

Editing a Dimension

Step 1 On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane. On the displayed page, click the **Dimensions** tab.

Step 2 In the dimension list, select the target dimension and click **Edit** in the **Operation** column.

Figure 5-96 Editing a dimension



Step 3 Edit the dimension information based on service requirements. For details about how to set parameters, see **Dimension parameters**.

Step 4 Click **Save**. Alternatively, click **Publish** to publish the edited dimension.

NOTE

You can choose to publish the dimension to the production or development environment. By default, it is published to the production environment. If you do not choose an environment, the dimension cannot be published.

----End

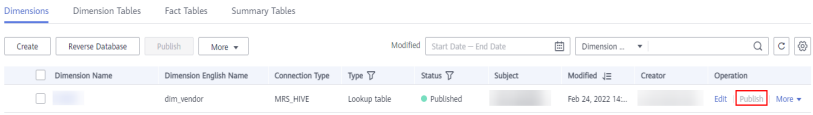
Publishing a Dimension

If a dimension is created but not published, perform the following steps to publish the dimension:

Step 1 On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane. On the displayed page, click the **Dimensions** tab.

Step 2 In the dimension list, find the target dimension and click **Publish** in the **Operation** column.

Figure 5-97 Publishing a dimension



Step 3 In the dialog box displayed, select a reviewer and click **OK**.

NOTE

You can choose to publish the dimension to the production or development environment. By default, it is published to the production environment. If you do not choose an environment, the dimension cannot be published.

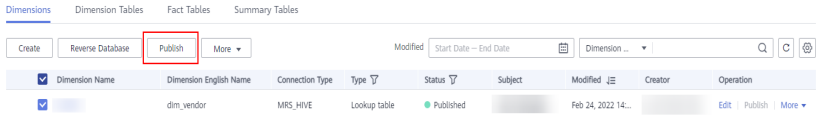
----End

You can also perform the following steps to publish multiple dimensions:

Step 1 On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane. On the displayed page, click the **Dimensions** tab.

Step 2 Select the dimensions you want to publish and click **Publish** above the dimension list.

Figure 5-98 Publishing multiple dimensions



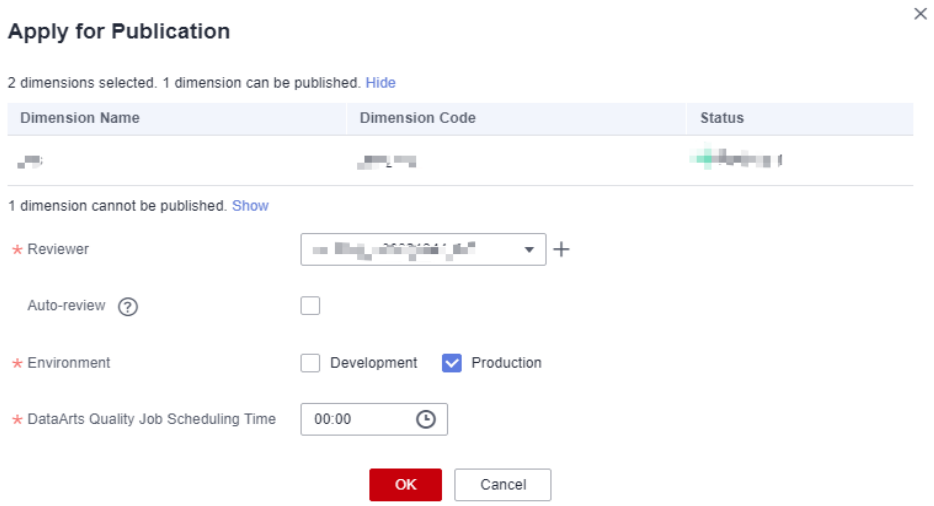
Step 3 In the displayed dialog box, select a reviewer, set **Job Scheduling Time**, and click **OK**.

NOTE

You can choose to publish the dimensions to the production or development environment. By default, they are published to the production environment. If you do not choose an environment, the dimensions cannot be published.

Job Scheduling Time refers to the scheduling time for automatic quality job creation after the dimension is published.

Figure 5-99 Publishing multiple dimensions



----End

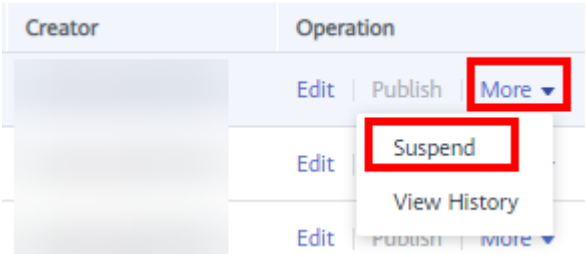
Suspending a Dimension

To suspend a published dimension, perform the following steps:

Step 1 On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane. On the displayed page, click the **Dimensions** tab.

Step 2 In the dimension list, find the target dimension and choose **More** > **Suspend** in the **Operation** column.

Figure 5-100 Suspending a dimension



Step 3 In the dialog box displayed, select a reviewer and click **OK**. The dimension is suspended after the reviewer approves it.

----End

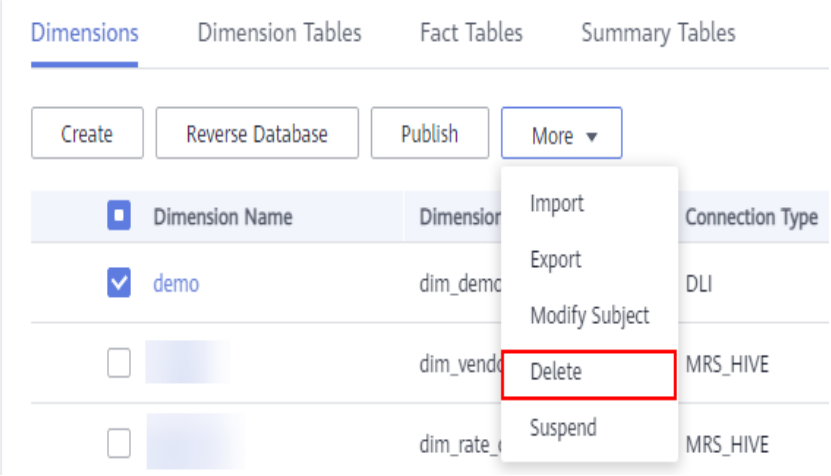
Deleting a Dimension

If a dimension is no longer needed, you can delete it. However, if the dimension has been published, you must suspend the dimension before deleting it. For details, see **Suspending a Dimension**.

Step 1 On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane. On the displayed page, click the **Dimensions** tab.

Step 2 In the dimension list, find the target dimension and choose **More** > **Delete** above the list.

Figure 5-101 Deleting a dimension



Step 3 In the **Delete Dimension** dialog box, confirm the information and click **Yes**.

If you select **Delete physical tables** in the dialog box, the physical tables in the database are also deleted when you delete the dimension.

----End

Importing a Dimension by Reversing a Database

By reversing databases, you can import one or more created database tables from other data sources into a dimension directory to turn them into dimensions.

Step 1 On the DataArts Architecture console, choose **Models** > **Dimensional Modeling** in the navigation pane on the left.

Step 2 Above the dimension list, click **Reverse Database**.

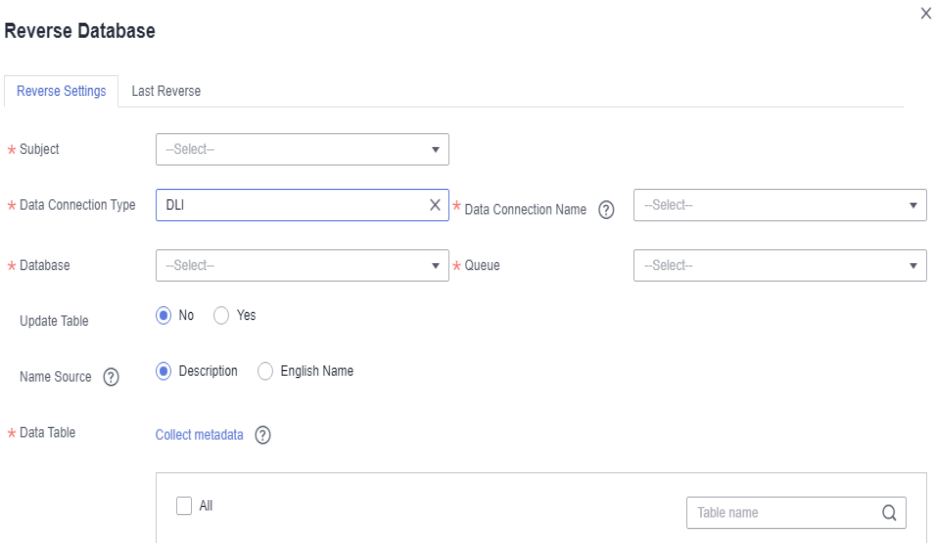
Step 3 In the displayed dialog box, set required parameters and click **OK**.

Table 5-37 Parameters for reversing the database

Parameter	Description
*Subject	Select a subject from the drop-down list.
*Data Connection Type	The data connection types supported by the reverse database are displayed in the drop-down list box. Select the required data connection type.

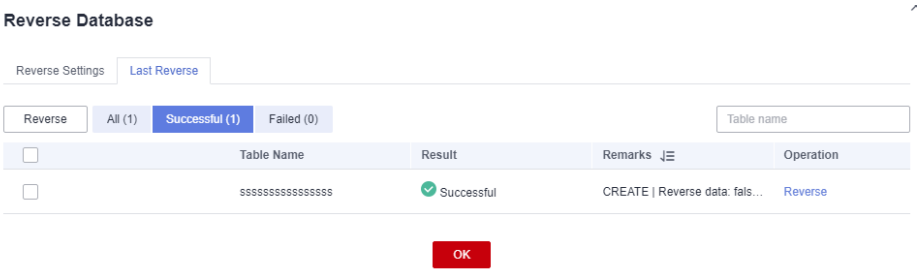
Parameter	Description
*Data Connection Name	Select a data connection. If you want to reverse a database from other data sources to a dimension directory, you must create a data connection in Management Center to connect to the data source. For details on how to create data connections, see **Managing Data Connections**.
*Database	Select a database.
*Schema	Select a value from the drop-down list box. This parameter is available only for DWS and PostgreSQL tables.
Queue	DLI queue. This parameter is available only when **Data Connection Type** is set to **DLI**.
Update Table	When **Yes** is selected, if the name of the reversed table is the same as that of an existing table in the dimension, the existing dimension is updated.
*Data Table	You can select **All** or **Partial**.

Figure 5-102 Reverse Database dialog box



Step 4 You can view the result on the **Last Reverse** tab page. If the reverse operation is successful, click **Close**. If the reverse operation fails, you can view the failure cause. After the fault is rectified, select the table again and click **Reverse** to retry.

Figure 5-103 Last Reverse tab page



----End

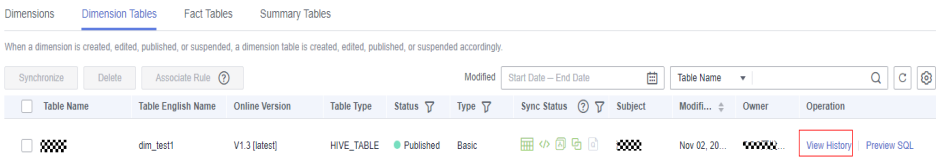
5.6.2.2 Managing Dimension Tables

A dimension table corresponds to a dimension and consists of a wide range of dimension fields. Creating, publishing, editing, and suspending a dimension table highly relate to the corresponding dimension. After a dimension is published, the system automatically creates and publishes the corresponding dimension table.

Viewing the Publish History of a Dimension Table

1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Dimension Tables** tab.
3. Select a dimension table in the list and click **View History** in the **Operation** column.

   Figure 5-104 Dimension Tables tab page

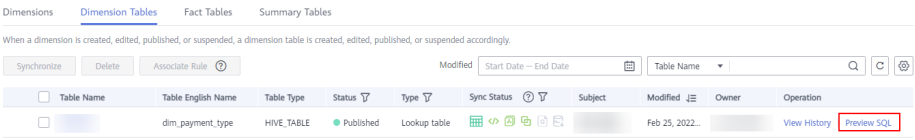
4. On the page displayed, you can view the publish history, version comparison information, and publish log of the dimension table.

   If the publish log includes error logs, the publishing has failed. You can click **Resynchronize** to synchronize the table to other DataArts Studio modules.

Previewing SQL

1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Dimension Tables** tab.
3. Select a dimension table in the list and click **Preview SQL** in the **Operation** column.

Figure 5-105 Previewing an SQL statement



4. On the page displayed, you can view or copy the SQL statement.

Synchronizing a Dimension Table

After you create or edit a dimension, you can manually synchronize the dimension table if the synchronization fails.

NOTE

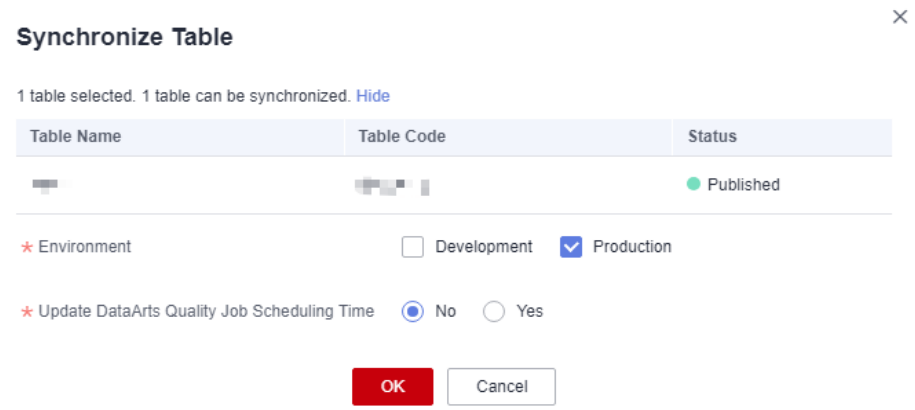
- The system performs the synchronization based on the data table update mode on the **Function Settings** tab page of **Configuration Center**. For details, see **Functions**.

1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Dimension Tables** tab.
3. In the dimension table list, select the target dimension table and click **Synchronize** above the list. The dialog box for synchronizing the dimension table is displayed.

   NOTE

   You can choose to synchronize the dimension tables to the production or development environment. By default, they are synchronized to the production environment. If you do not choose an environment, the tables cannot be synchronized.

Figure 5-106 Synchronizing dimension tables



4. After confirming that the information is correct, click **OK**. The synchronization result is displayed.

   After the synchronization, you can view the synchronization status of the dimension table in the dimension table list. You can also click  above the list to refresh the status. You can switch between the production environment and development environment to view the synchronization result.

Associating a Dimension Table with a Quality Rule

1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Dimension Tables** tab.
3. In the dimension table list, select the target dimension table, and click **Associate Rule**.

   Figure 5-107 Associating a dimension table with a quality rule

4. On the page displayed, set the parameters. After the configuration is complete, click **OK**.
   - **Update Existing Rule**: If this option is selected, the newly added rule will overwrite the old rule.
   - **Table Field**: This parameter applies to all fields by default. You can enter a regular expression to filter fields as required.
   - **WHERE Clause**: This parameter can be used to filter fields.
   - **Generate Anomaly Data**: If this option is enabled, anomaly data is stored in the specified database based on the configured parameters.
   - **Database/Schema**: database or schema that stores anomaly data. This parameter is displayed when **Generate Anomaly Data** is enabled.
   - **Table Prefix**: prefix of the table that stores anomaly data. This parameter is displayed when **Generate Anomaly Data** is enabled.
   - **Table Suffix**: suffix of the table that stores anomaly data. This parameter is displayed when **Generate Anomaly Data** is enabled.
   - **Add Rule**: You can click **Add Rule** to add a rule. For example, add a rule named **Unique value**, select the rule, click **OK**, enter an alarm condition expression in the **Alarm Condition** text box, add other rules in the same way, and click **OK**.
   - An alarm condition expression consists of alarm parameters and logical operators. When a quality job is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered. In the **Associate Quality Rule** dialog box, the alarm parameters of each quality rule are displayed as buttons.

Associating a Single Field with a Quality Rule

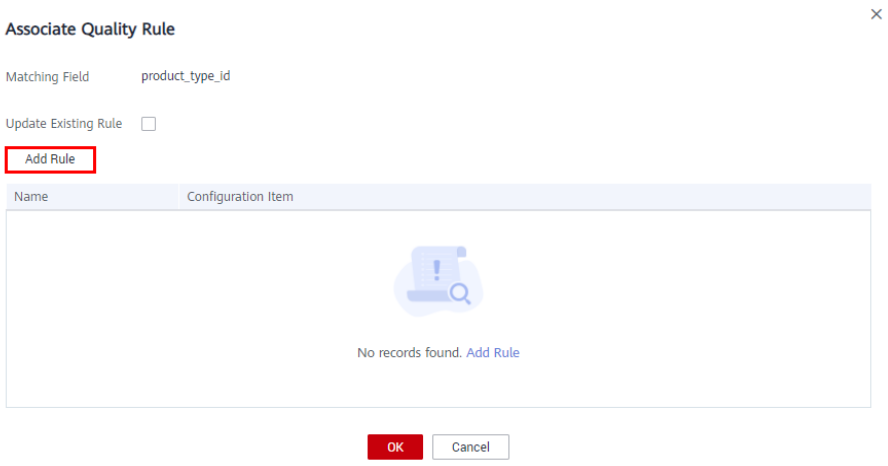
1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Dimension Tables** tab.
3. In the dimension table list, click the name of the target dimension table.
4. In the field list on the dimension table details page, click in the row of the target field to associate the field with a quality rule.

   Figure 5-108 Associating a single field with a quality rule

5. After the configuration is complete, click **OK**.
   - **Update Existing Rule**: If this option is selected, the newly added rule will overwrite the old rule.
   - **Add Rule**: You can click **Add Rule** to add a rule. For example, add a rule named **Unique value**, select the rule, click **OK**, enter an alarm condition expression in the **Alarm Condition** text box, add other rules in the same way, and click **OK**.
   - An alarm condition expression consists of alarm parameters and logical operators. When a quality job is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered. In the **Associate Quality Rule** dialog box, the alarm parameters of each quality rule are displayed as buttons.

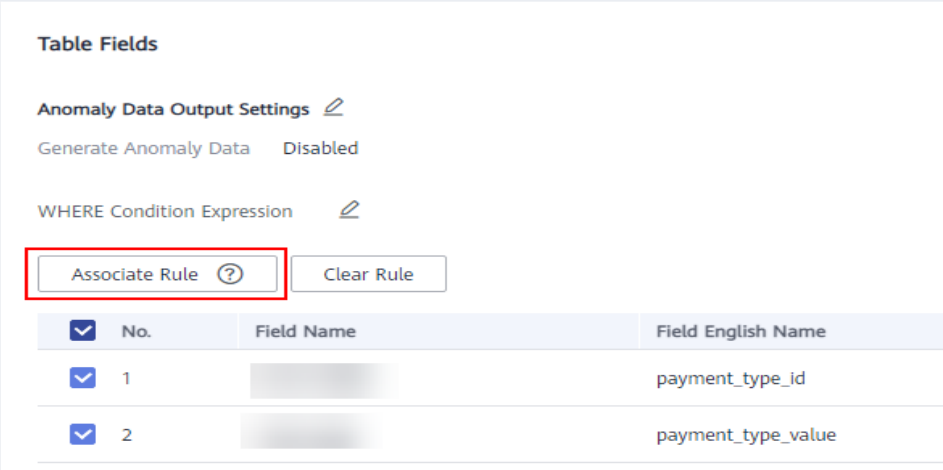
   Figure 5-109 Adding a rule

Associating Table Fields with a Quality Rule in Batches

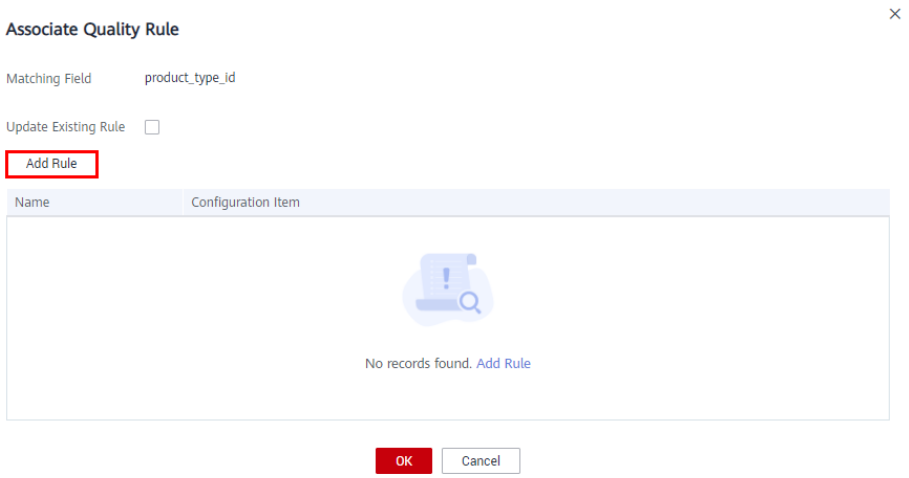
1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Dimension Tables** tab.
3. In the dimension table list, click the name of the target dimension table.
4. In the table field list on the dimension table details page, select the target table fields and click **Associate Rule**.

   Figure 5-110 Associating table fields with a quality rule

5. On the page displayed, add a rule and set the rule parameters.
   - **Update Existing Rule**: If this option is selected, the newly added rule will overwrite the old rule.
   - **Add Rule**: You can click **Add Rule** to add a rule. For example, add a rule named **Unique value**, select the rule, click **OK**, enter an alarm condition expression in the **Alarm Condition** text box, add other rules in the same way, and click **OK**.
   - An alarm condition expression consists of alarm parameters and logical operators. When a quality job is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered. In the **Associate Quality Rule** dialog box, the alarm parameters of each quality rule are displayed as buttons.

Figure 5-111 Associating table fields with a quality rule

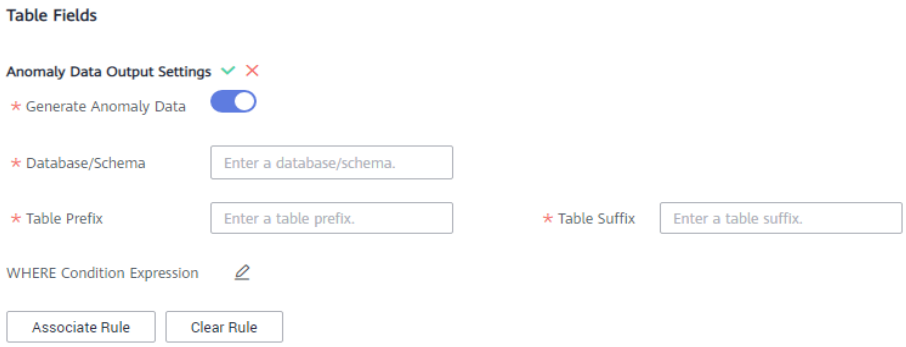


6. (Optional) If you want to store anomaly data that does not comply with the preset rules in the exception table, enable **Anomaly Data Output Settings**.

Figure 5-112 Enabling Anomaly Data Output Settings

Click the pen icon next to **Anomaly Data Output Settings** and enable **Generate Anomaly Data**. The anomaly data will be stored in the specified database based on the settings.

Figure 5-113 Anomaly Data Output Settings



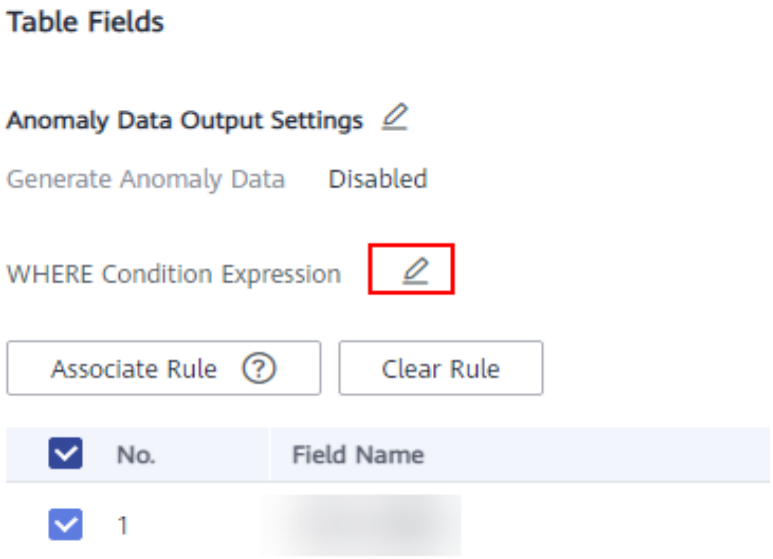
The parameters are as follows:

- **Database/Schema**: database or schema that stores anomaly data
- **Table Prefix**: prefix of the table that stores anomaly data
- **Table Suffix**: suffix of the table that stores anomaly data

Click ✓ to save the settings.

7. (Optional) By default, the quality rule applies to the entire table. If you want to query data in specified partitions, set the where condition.

   Figure 5-114 Where condition



8. After the configuration is complete, click **OK**.

Deleting a Dimension Table

Dimensions in publishing review, published, or suspension review state cannot be deleted. You can delete a dimension table on the **Dimensions** page.

1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Dimension Tables** tab.
3. In the dimension table list, select the target dimension table and click **Delete** above the list.

   Figure 5-115 Deleting a dimension table

4. Confirm the dimension table to delete, and click **Yes**.

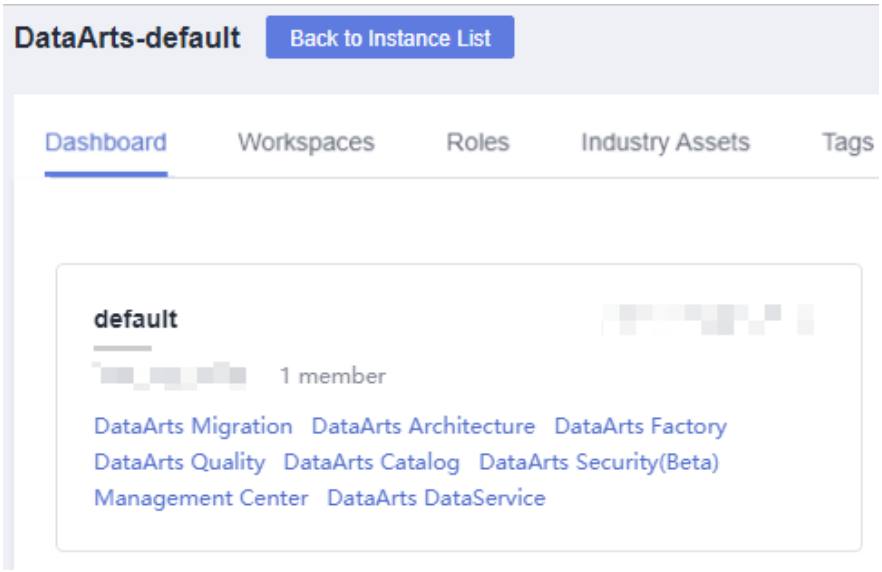
5.6.2.3 Creating Fact Tables

A fact table for a business process can provide a wealth of information about specific business processes. After a fact table is created, the public affair details are accumulated to facilitate data extraction.

Creating and Publishing a Fact Table

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

   Figure 5-116 DataArts Architecture

2. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane. On the displayed page, click the **Fact Tables** tab.
3. Select a subject from the subject tree on the left and click **Create**.
4. On the **Create Fact Table** page, perform the following operations:
   a. Set the parameters in the **Basic Settings** area.

   Figure 5-117 Basic Settings area

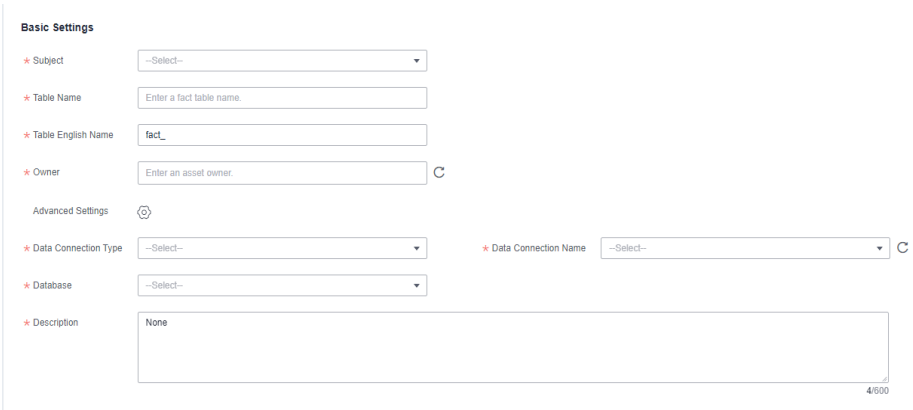
   

Table 5-38 Parameters in the Basic Settings area

Parameter	Description
*Subject	Select a subject (business domain group > business domain > business object) where you can place the fact table.
*Table Name	It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_
*Table English Name	Table codes must start with letters. Only letters, numbers, and underscores (_) are allowed.
*Data Connection Type	Select a data connection type from the drop-down list box.
*Data Connection Name	Select a data connection from the drop-down list box. It is recommended that the same data connection be used for dimension modeling.
*Database	Select a database from the drop-down list box.
Queue	DLI queue. This parameter is displayed only for DLI data connections.
Schema	DWS or POSTGRESQL mode. This parameter is displayed only for DWS and POSTGRESQL data connections.

Parameter	Description
Table Type	DLI models support the following table types: • **MANAGED**: Data is stored in a DLI table. • **EXTERNAL**: Data is stored in an OBS table. When **Table Type** is set to **EXTERNAL**, you must set **OBS Path**. The OBS path format is ***/bucket_name/filepath***. DWS models support the following table types: • **DWS_ROW**: row-store table. Tables are stored to disk partitions by row. • **DWS_COLUMN**: column-store table. Tables are stored to disk partitions by column. • **DWS_VIEW**: view-store table. Tables are stored to disk partitions by view. The MRS Hive model supports **HIVE_TABLE** and **HIVE_EXTERNAL_TABLE**. The MRS Spark model supports **HUDI_COW** and **HUDI_MOR**. The PostgreSQL model supports only **POSTGRESQL_TABLE**. The MRS_CLICKHOUSE model supports only **CLICKHOUSE_TABLE**. The Oracle model supports only **ORACLE_TABLE**. The MySQL model supports only **MYSQL_TABLE**.
Compression Level	This parameter is available when the data connection type is DWS. The following compression levels are available for different table types: • **DWS_ROW**: **NO** and **YES** • **DWS_COLUMN**: **NO**, **LOW**, **MIDDLE**, and **HIGH**. • **DWS_VIEW**: The compression level is not supported.

Parameter	Description
Distributed By	This parameter is displayed only for DWS data connections. You must add a table field before selecting a table field from the drop-down list as a **Distributed By** field. Multiple table fields can be selected. Currently, only **REPLICATION** and **HASH** are supported. ● **REPLICATION**: A full table is stored on each DN. The advantage of this option is that each DN has all the data of a table. During the join operation, data redistribution can be avoided, reducing network overhead. The disadvantage is that each DN retains the complete data of a table, resulting in data redundancy. Generally, this option is recommended for small dimension tables. ● **HASH**: If you select this option, you must specify a distribution key for the user table. When a record is inserted, the system performs hash computing based on values in the distribute keys and then stores data on the corresponding DN. In a Hash table, I/O resources on each node can be used during data read/write, which improves the read/write speed of a table. Generally, this option is recommended for large dimension tables (a large dimension table contains over 1 million records).
PreCombineField	This parameter is available only for Spark data connections.
Path	This parameter is available only when the data source is MRS Hive and table type is HIVE_EXTERNAL_TABLE. The path can contain only letters, digits, slashes (/), periods (.), hyphens (-), underscores (_), and colons (:).
*Owner	You can enter an owner name or select an existing owner.
Advanced Settings	Set custom items to describe the table. The custom items can be viewed in the table details. For example, if you want to identify the source of the table, you can add item **source** and set its value to the table source information. Then you can view the table source information in the table details.
*Description	A description of the fact table. It allows 1 to 600 characters.

b. On the **Field Settings** page, click **Create** and select **Dimension** or **Measure** to add a dimension or measure field.

   ▪ If you select **Dimension**, select one or multiple dimensions in the displayed dialog box and click **OK**

- If you select **Measure**, set required parameters to add a measure field.

For details about the field parameters, see **Table 5-39**. After adding a field, you can click ↑ or ↓ to move the field up or down.

Figure 5-118 Adding a dimension or measure field

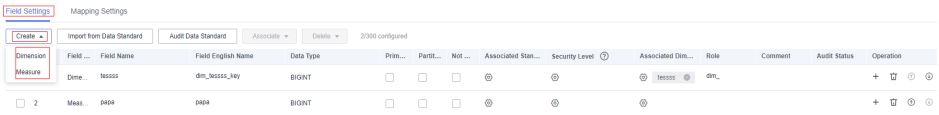


Table 5-39 Field parameters

Parameter	Description
Type	Two types are available: **Measure** and **Dimension**.
Name	It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_ The surrogate key name of the added dimension field is displayed automatically. Generally, you do not need to change the name.
Field Code	Table codes must start with letters. Only letters, numbers, and underscores (_) are allowed.
Data Type	Data type of the created dimension
Primary Key	If this parameter is selected, the field is a primary key. **NOTE** If an MRS Spark connection is used to connect to MRS Hudi data sources, data can be written to the database only if fields have primary keys. Otherwise, table synchronization fails.
Partition	If this parameter is selected, the field is a partition field.
Not Null	Whether the parameter value can be left empty.
Associate Standard	If you have created data standards, click ⚙ to select one to associate with the field. If **Create Data Quality Jobs** is selected for **Model Design Process** on the **Function Settings** tab page in Configuration Center and a field is associated with a data standard, a quality job is automatically generated after a table is published. A quality rule is generated for each field associated with the data standard. The quality of the field is monitored based on the data standard. You can access the **Quality Job** page of DataArts Quality to view the job details. If no data standard is available, create one. See **Creating Data Standards** for details.

Parameter	Description
Security Level	You can click to add a security level for the logical entity attribute. If you cannot find the security level you want, click **go to** to go to the DataArts Security console and create a security level. You can disable this function on the **Models** tab page on the **Configuration Center** page.
Associate Dimension	Only dimension fields need to be associated with dimensions. The name of the associated dimension. Click to replace the associated dimension. If the public workspace is enabled, you can select the public workspace dimension.
Role	Roles need to be assigned to dimension fields which are added for multiple times. This is not required for measure fields. If a dimension is added multiple times, set different roles to distinguish the dimensions.
Description	A description of the dimension.
Audit Status	Whether to audit the data standard
Operation	Related operations

c. On the **Mapping Settings** tab page, click **Create Mapping** and set mapping parameters.

Figure 5-119 Configuring mapping parameters

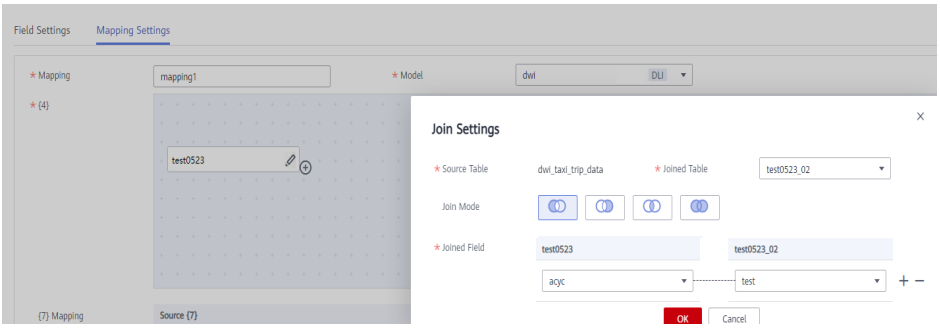
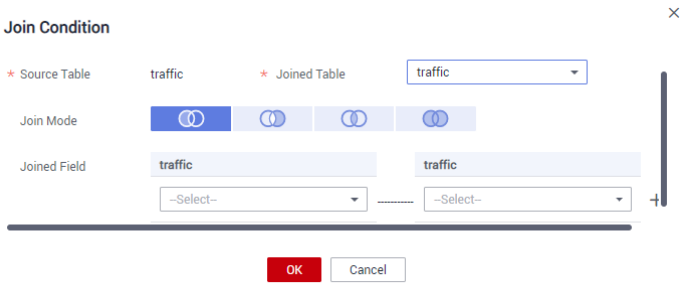


Table 5-40 Parameters of mappings

Parameter	Description
*Mapping	Only letters, numbers, and underscores (_) are allowed.
*Model	Select a created relationship model from the drop-down list box. If no relationship model has been created, create one. See **Designing Physical Models**.
*Table	Select a table from which data is obtained. If data is obtained from multiple tables, click next to the table name to set the JOIN condition between the table and other tables. 1. Select a JOIN mode. The JOIN mode includes left JOIN, right JOIN, inner JOIN, and outer JOIN from left to right. 2. Set the JOIN condition in the JOIN field. Generally, select the fields with the same meaning in the source table and joined table. Click or to add or delete a JOIN condition. The relationship between JOIN conditions is AND. 3. Click **OK**. 4. If you want to delete a joined table after setting the JOIN condition, click next to the table name. **Figure 5-120** Join Condition dialog box 
Field Mapping	Select a source field with the same meaning as the current mapping field. If a table field comes from multiple models, you must create multiple mappings. In each mapping, you only need to set the source field for the field that comes from the current mapping. Other fields do not need to be set.

5. Click **Publish**.

NOTE

You can choose to publish the fact table to the production or development environment. By default, it is published to the production environment. If you do not choose an environment, the fact table cannot be published.

6. Wait for the reviewer to approve the fact table.

   After the fact table is approved, it is automatically created in the database.

7. Go back to the fact table list and locate the table just published. View its synchronization status in the **Sync Status** column. You can switch between the production environment and development environment to view the synchronization result.
   - If the synchronization is successful, the fact table is successfully published and created in the database.
   - If the synchronization failed, choose **More** > **View History** in the row where the fact table is located. On the page displayed, click the **History** tab to view logs. Troubleshoot the problem based on the logs. After the error is rectified, choose **More** > **Synchronize** above the fact table list to issue the synchronization command again. If the problem persists, contact technical support personnel.

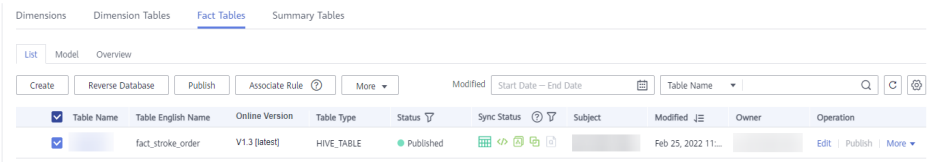
     NOTE

     You can choose to synchronize the fact table to the production or development environment. By default, it is synchronized to the production environment. If you do not choose an environment, the fact table cannot be synchronized.

Managing a Fact Table

After a fact table is created, you can access the **Fact Tables** page of **Dimensional Modeling** in DataArts Architecture. On the page displayed, you can edit, publish, suspend, and delete the fact table, as well as view the publish logs.

Figure 5-121 Fact table management



- **Editing a fact table**
  a. In the fact table list, select a fact table and click **Edit** to the right of it. The page for editing the fact table is displayed.
  b. Edit the table as required.
  c. Click **Save** to save the settings, or click **Publish** to publish the settings.

     NOTE

     You can choose to publish the fact table to the production or development environment. By default, it is published to the production environment. If you do not choose an environment, the fact table cannot be published.

- **Publishing a fact table**
  a. In the fact table list, select a fact table and click **Publish**. The dialog box for publishing the fact table is displayed.
  b. Select a reviewer from the drop-down list.

     NOTE

     You can choose to publish the fact table to the production or development environment. By default, it is published to the production environment. If you do not choose an environment, the fact table cannot be published.

  c. Click **OK**.
- **Viewing the publish history**
  a. Select a fact table in the list and choose **More** > **View History** on the right.
  b. On the page displayed, you can view the publish history and version comparison information of the fact table.

     If the publish log includes error logs, the publishing has failed. You can click **Resynchronize** to synchronize the table to other DataArts Studio modules.
- **Associating a fact table with a quality rule**
  a. Select a fact table in the fact table list and click **Associate Rule** above the list.
  b. In the **Associate Quality Rule** dialog box, you can add rules to the fields in the fact table in batches and associate the rules with the fields.
  c. Click **OK**.
- **Previewing an SQL statement**
  a. Select a fact table in the list and choose **More** > **Preview SQL** on the right.
  b. On the page displayed, you can view or copy the SQL statement.
- **Create a conversion**
  a. **DWS_COLUMN**: **NO**, **LOW**, **MIDDLE**, and **HIGH**.
  b. For details about how to create a derivative metric, see **Creating and Publishing a Derivative Metric**.
- **Suspending a fact table**
  a. In the fact table list, select a fact table and click **Suspend**. The dialog box for suspending a fact table is displayed.
  b. Select a reviewer from the drop-down list box.
  c. Click **OK**.

     NOTE

     - You can suspend or delete a fact table only when it is not referenced. For example, a fact table can be deleted only when it is not used by atomic metrics.
- **Deleting a fact table**

  If you no longer need a fact table, you can delete it. Fact tables in publishing review, published, or suspension review state cannot be deleted.

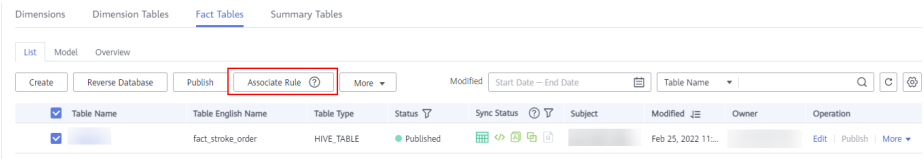
a. In the fact table list, select a fact table and choose **More** > **Delete** above the list.

b. In the dialog box displayed, click **Yes**.

Associating a Fact Table with a Quality Rule

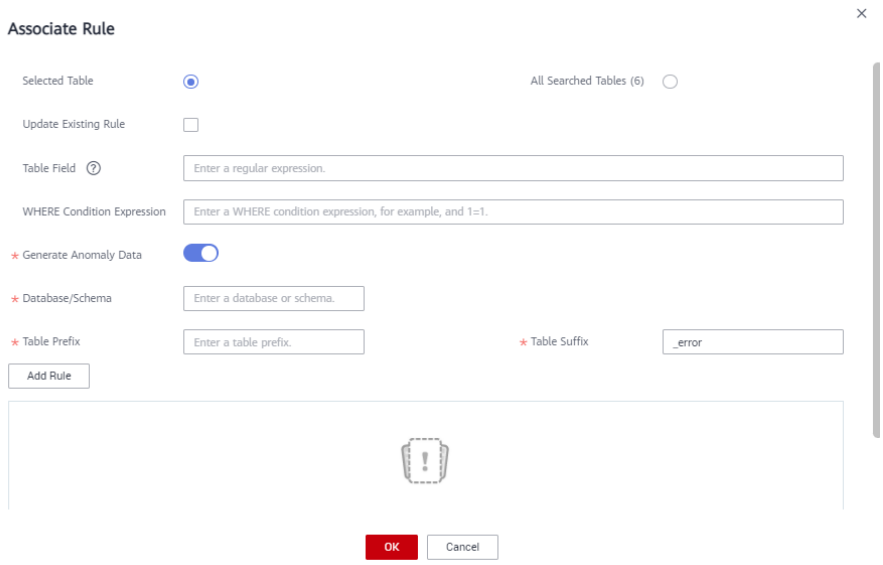
1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Fact Tables** tab.
3. In the fact table list, click the name of the target fact table. Click **Associate Rule**.

   Figure 5-122 Associating a fact table with a quality rule

4. On the page displayed, set the parameters. After the configuration is complete, click **OK**.
   - **Update Existing Rule**: If this option is selected, the newly added rule will overwrite the old rule.
   - **Table Field**: This parameter applies to all fields by default. You can enter a regular expression to filter fields as required.
   - **WHERE Clause**: This parameter can be used to filter fields.
   - **Generate Anomaly Data**: If this option is enabled, anomaly data is stored in the specified database based on the configured parameters.
   - **Database/Schema**: database or schema that stores anomaly data. This parameter is displayed when **Generate Anomaly Data** is enabled.
   - **Table Prefix**: prefix of the table that stores anomaly data. This parameter is displayed when **Generate Anomaly Data** is enabled.
   - **Table Prefix**: prefix of the table that stores anomaly data. This parameter is displayed when **Generate Anomaly Data** is enabled.
   - **Add Rule**: You can click **Add Rule** to add a rule. For example, add a rule named **Unique value**, select the rule, click **OK**, enter an alarm condition expression in the **Alarm Condition** text box, add other rules in the same way, and click **OK**.
   - An alarm condition expression consists of alarm parameters and logical operators. When a quality job is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered. In the **Associate Quality Rule** dialog box, the alarm parameters of each quality rule are displayed as buttons.

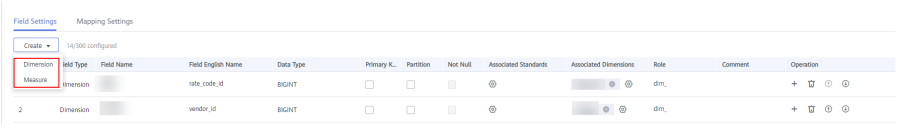
Figure 5-123 Associating a fact table with a quality rule



Creating a Field in the Fact Table

1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Fact Tables** tab.
3. In the fact table list, click the name of the target table and click **Edit** in the **Operation** column.
4. Click **Create** in the **Table Fields** area, select a new field type from the drop-down list, and set the related parameters.

   Figure 5-124 Creating a field

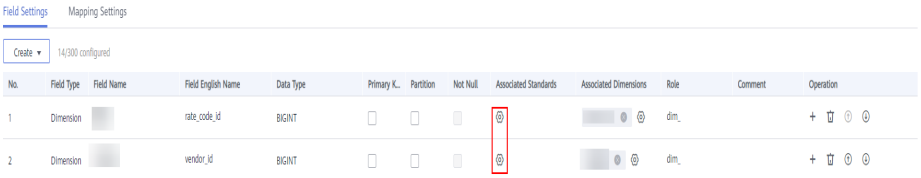
   

5. After the configuration is complete, click **OK**.

Associating a Fact Table Field with a Data Standard

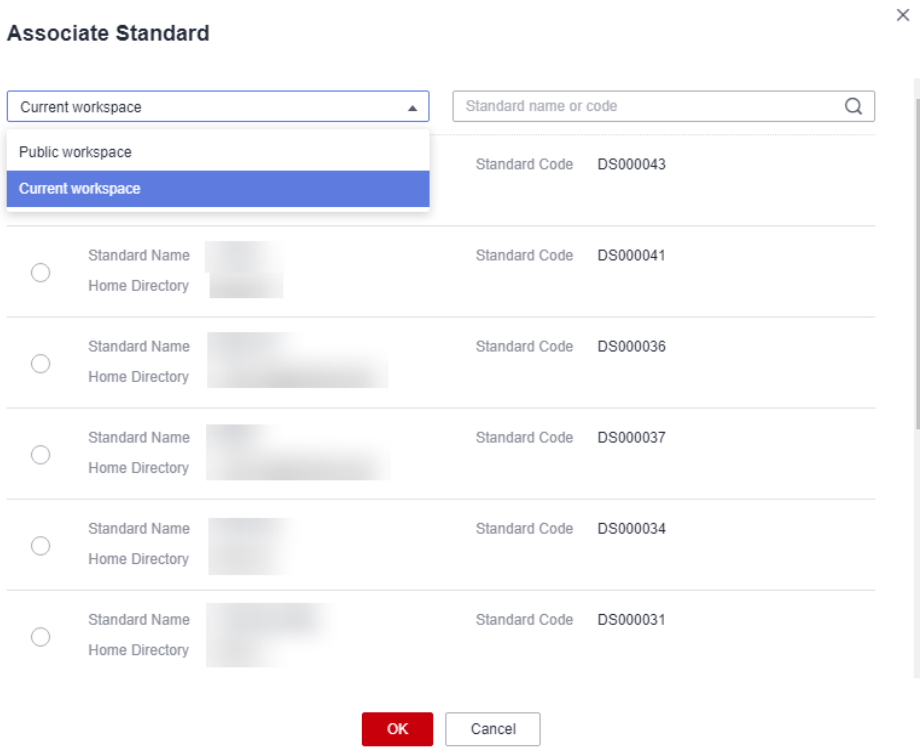
1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Fact Tables** tab.
3. Click the name of the target fact table in the list.
4. In the table field list on the details page of the fact table, search for the target field, click corresponding to the field to configure the association between the field and the data standard. For details on the sources of data standards, see **Creating a Data Standard**.

Figure 5-125 Associating a fact table field with a data standard



5. After the configuration is complete, click **OK**. If a public workspace is available, you need to manually set the data standard source to the public workspace or the current workspace when selecting a data standard in a common workspace. When **Public workspace** is enabled, the data standards of the public workspace can be referenced in common workspaces.

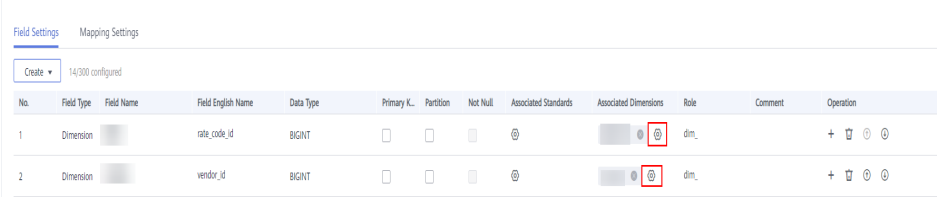
Figure 5-126 Associating a data standard



Associating a Fact Table Field with a Quality Rule

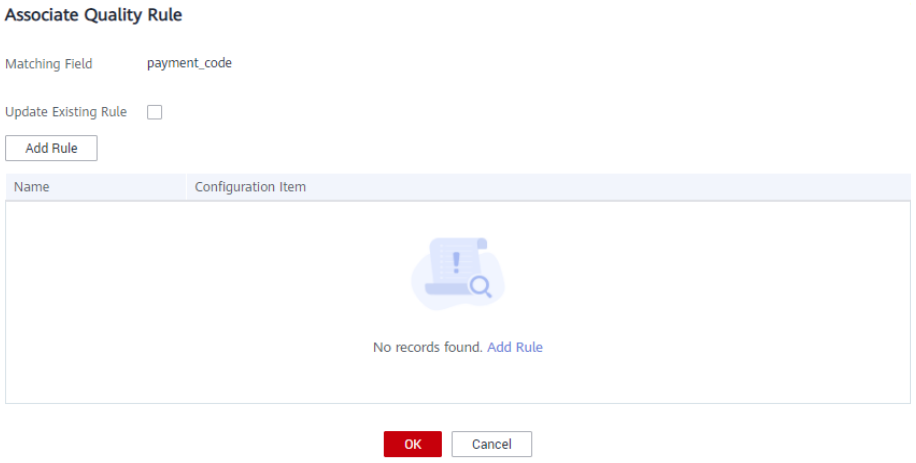
1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Fact Tables** tab.
3. In the fact table list, click the name of the target fact table.
4. In the table field list on the fact table details page, locate the target field and click to associate the field with a quality rule.

Figure 5-127 Associating a fact table field with a quality rule



5. After the configuration is complete, click **OK**.

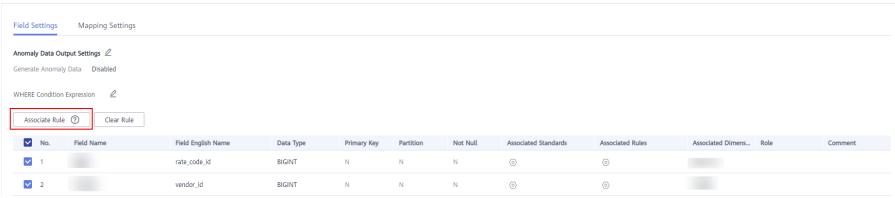
Figure 5-128 Adding a rule



Associating Fact Table Fields with a Quality Rule in Batches

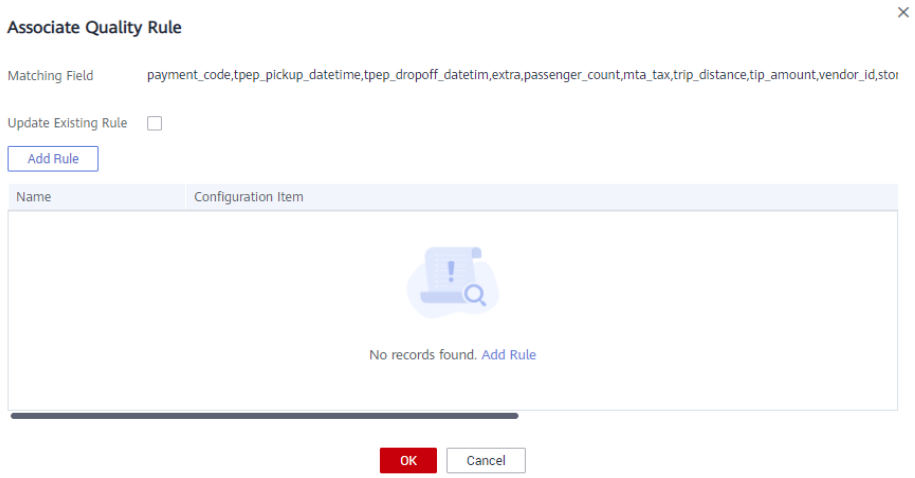
1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Fact Tables** tab.
3. In the fact table list, click the name of the target fact table.
4. In the table field list on the fact table details page, select the target table fields and click **Associate Rule**.

Figure 5-129 Associating fact table fields with a quality rule



5. On the page displayed, add a rule and set the rule parameters.

Figure 5-130 Adding a rule



6. After the configuration is complete, click **OK**.

Importing a Fact Table by Reversing a Database

By reversing databases, you can import one or more created database tables from other data sources into a fact table directory to turn them into fact tables.

Step 1 On the DataArts Architecture console, choose **Models** > **Dimensional Modeling** in the navigation pane on the left.

Step 2 Above the fact table list, click **Reverse Database**.

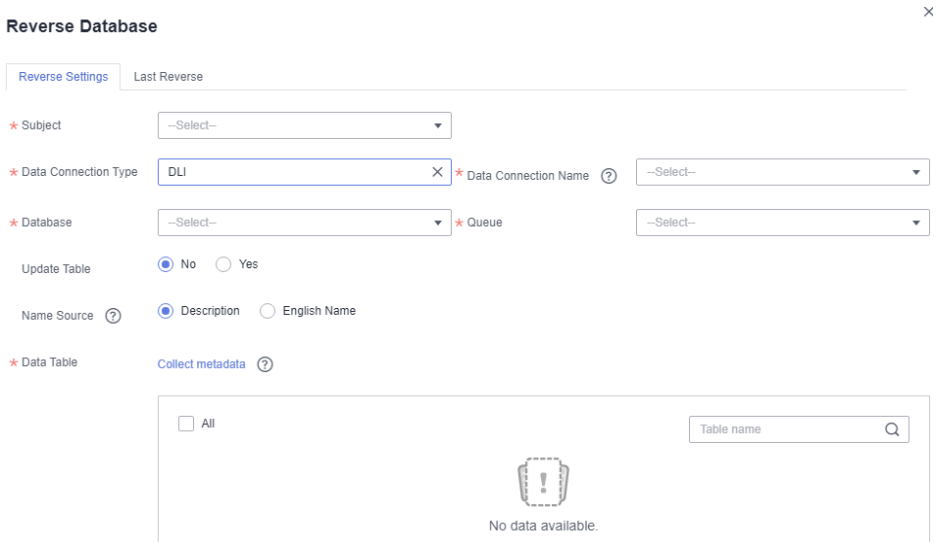
Step 3 In the displayed dialog box, set required parameters and click **OK**.

Table 5-41 Parameters for reversing the database

Parameter	Description
*Subject	Select a subject from the drop-down list.
*Data Connection Type	The data connection types supported by the reverse database are displayed in the drop-down list box. Select the required data connection type.
*Data Connection Name	Select a data connection. If you want to reverse a database from other data sources to a fact table directory, you must create a data connection in Management Center to connect to the data source. For details on how to create data connections, see **Managing Data Connections**.
*Database	Select a database.
*Schema	Select a value from the drop-down list box. This parameter is available only for DWS and PostgreSQL tables.
Queue	DLI queue. This parameter is available only when **Data Connection Type** is set to **DLI**.

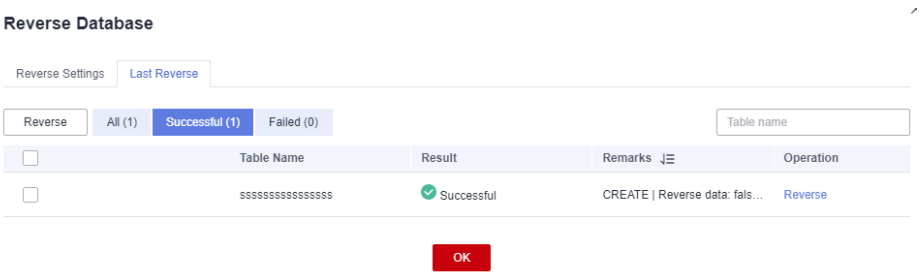
Parameter	Description
Update Table	When **Yes** is selected, if the name of the reversed table is the same as that of an existing fact table, the existing fact table is updated.
*Data Table	You can select **All** or **Partial**.

Figure 5-131 Reverse Database dialog box



Step 4 You can view the result on the **Last Reverse** tab page. If the reverse operation is successful, click **Close**. If the reverse operation fails, you can view the failure cause. After the fault is rectified, select the table again and click **Reverse** to retry.

Figure 5-132 Last Reverse tab page



----End

5.7 Metric Design

5.7.1 Business Metrics

After data survey and requirement analysis, you must implement metrics. A metric is a statistical value that measures the overall characteristic of a target and reflects the business situation in a business activity of an enterprise. A metric consists of its name and value. The metric name and its definition reflect the quality and quantity of the metric. The metric value reflects the quantifiable values of the specified time, location, and condition of the metric.

Business metrics define the purposes and calculation formulas of technical metrics and are not used to perform actual calculation. Business metrics can be associated with technical metrics. Technical metrics implement business metrics and define calculation methods.

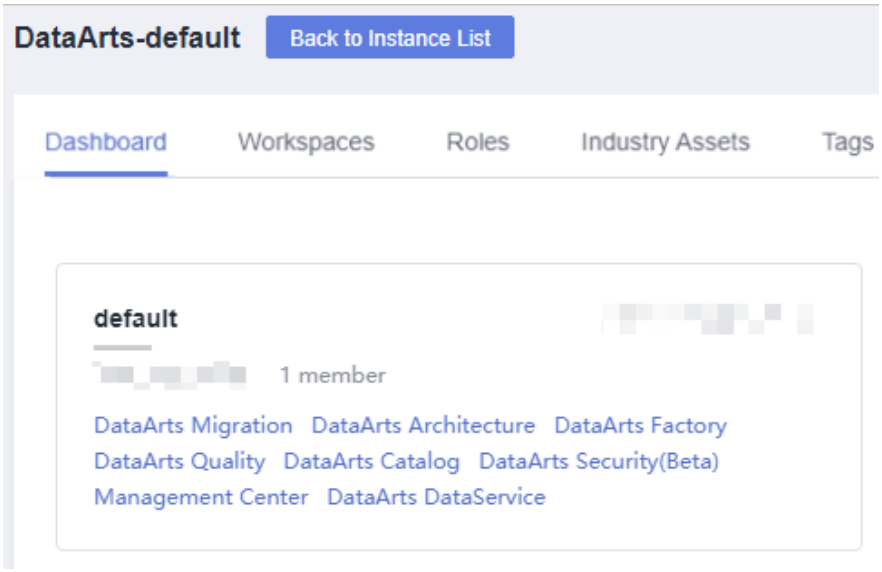
Prerequisites

You have designed a process. For details, see **Designing Processes**.

Creating and Publishing a Business Metric

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

   Figure 5-133 DataArts Architecture

2. On the DataArts Architecture page, choose **Metrics** > **Business Metrics** in the left navigation pane.
3. In the process tree on the left, select a process and click **Create**.
4. On the page displayed, set the parameters and click **Publish**.
   a. Configure basic settings.

Figure 5-134 Basic Settings area

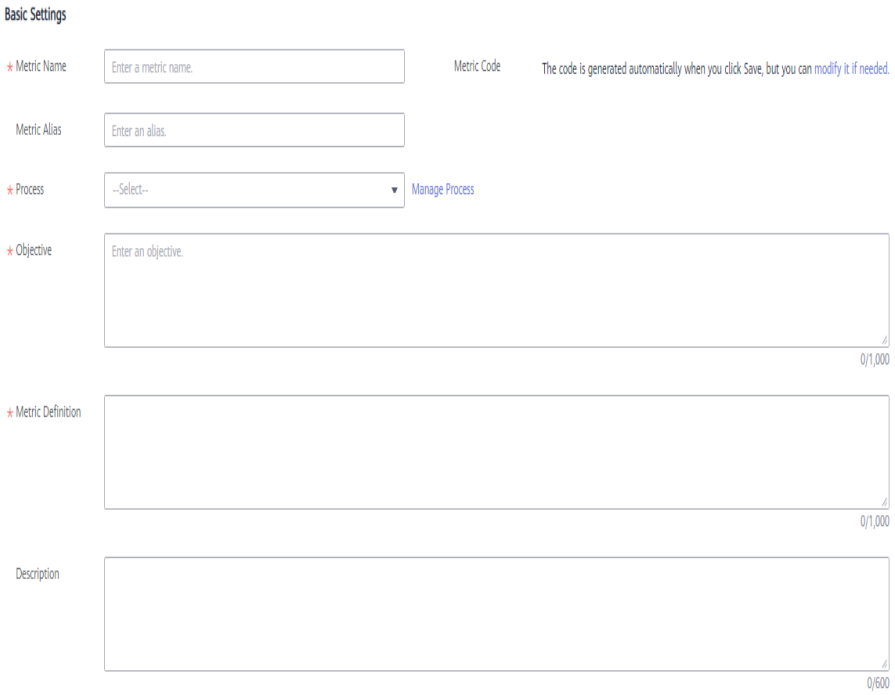


Table 5-42 Parameters in the Basic Settings area

Parameter	Description
*Metric Name	The name of the business metric to create. The value can contain only letters, digits, underscores (_), hyphens (-), brackets, commas (,), spaces, and the following special characters: +#[]/. It cannot start or end with a space.
Code	• The metric code is automatically generated. You can configure the generation rule on the **Configuration Center** page of DataArts Architecture. For details, see **Encoding Rules**.
Alias	This parameter is optional.
*Process	Select the process that the metric belongs to. If no process is available, create one. Refer to **Designing Processes** for details.
*Objective	Your purpose of setting the metric.
*Metric Definition	The definition of the metric must be accurately described.
Remarks	Remarks for the metric to create.

Parameter	Description
Custom metric	If a custom metric is configured on the **Metrics** tab page of the configuration center, the metric is displayed as a parameter on this page. For how to create a custom field, see **Metric Settings**.

b. Configure the metric information.

Figure 5-135 Metric Information area

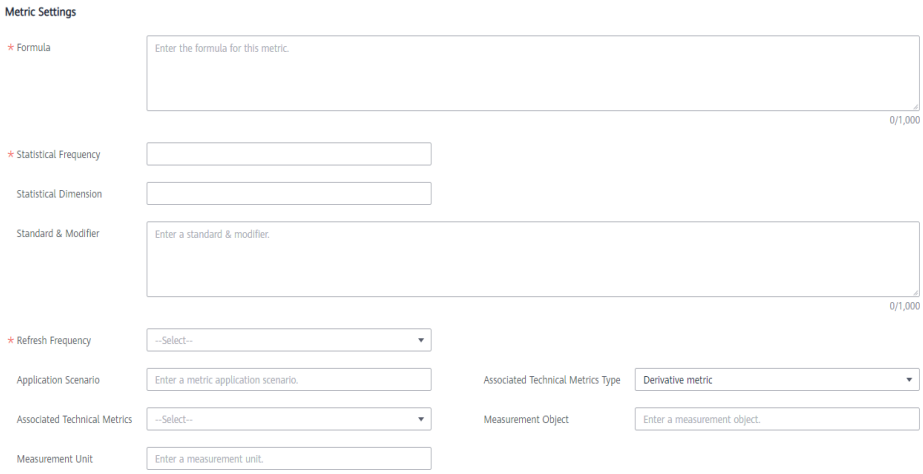


Table 5-43 Parameters in the Metric Information area

Parameter	Description
*Formula	The computing logic of the business metric, which guides developers to design atomic and derivative metrics. Business metrics are used to guide the implementation of technical metrics only and are not calculated.
*Statistical Frequency	The statistical period of a metric, which helps developers set the time limits.
Statistical Dimension	You can select an existing dimension from the drop-down list For details on how to create a dimension, see **Creating Dimensions**.
Standard & Modifier	Modifiers are abstract definitions of scenarios and are used to determine the measurement scope.
*Refresh Frequency	The interval for updating a metric. Developers or operators can set the scheduling frequency of derivative metrics based on the metric update frequency.

Parameter	Description
Metric Application Scenario	The application scenarios of the metric.
Associated Technical Metrics Type	Select the type of the technical metric associated with the business metric. Available options include **Derivative metric**, **Compound metric**, and **Atomic metric**.
Associated Technical Metrics	Select a technical metric associated with the business metric.
Measurement Object	The field for measuring a metric.
Measurement Unit	The measurement unit of a metric.

c. Configure the management information.

Figure 5-136 Management Information area

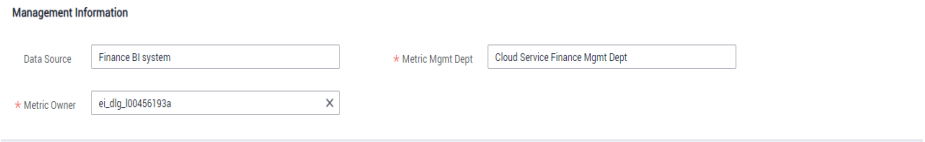


Table 5-44 Parameters in Management Information area

Parameter	Description
Data Source	The generator of data.
*Metric Mgmt Dept	The department that manages the metric.
*Metric Owner	Metric owner. You can enter the name of an owner or select an existing owner.

5. In the dialog box displayed, select a reviewer and click **OK**.
6. Repeat **3** to **5** to create and publish other business metrics.
7. All the business metrics must be approved by reviewers.

   If the applications are approved, the business metrics are created.

   Click the name of a business metric to view its details, relationship diagram, publishing history, and review history.

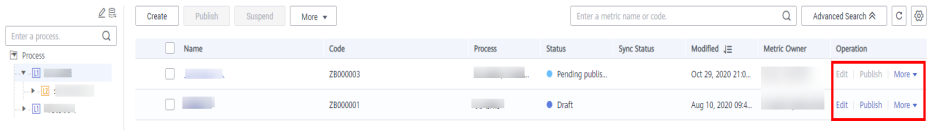
In the relationship diagram, you can view the lineage diagram of the business metric.

In the release history, you can view the differences between historical versions.

Editing a Business Metric

1. On the DataArts Architecture page, choose **Metrics** > **Business Metrics** in the left navigation pane.

   Figure 5-137 Managing business metrics

2. In the business metric list, select the target metric and click **Edit** on the right.
3. Edit the business metric information as required.
4. Click **Save** to save the settings. Alternatively, click **Publish** to publish the edited business metric.

Publishing a Business Metric

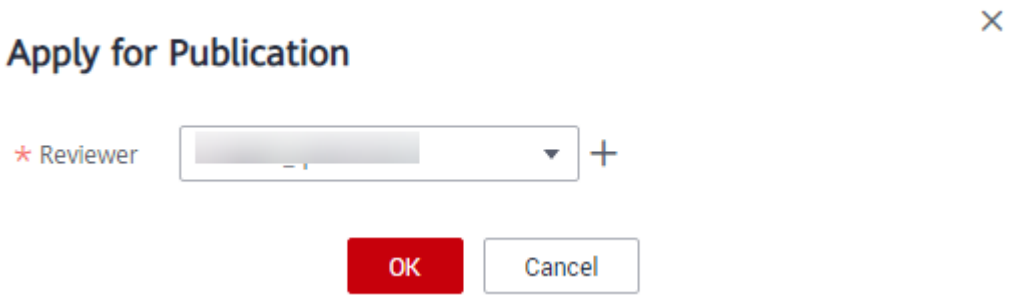
If a business metric is created but not published, perform the following steps to publish it:

Step 1 On the DataArts Architecture page, choose **Metrics** > **Business Metrics** in the left navigation pane.

Step 2 In the business metric list, select the target metric and click **Publish**.

Step 3 In the dialog box displayed, select a reviewer and click **OK**.

Figure 5-138 Submit for Publication dialog box



----End

Suspending a Business Metric

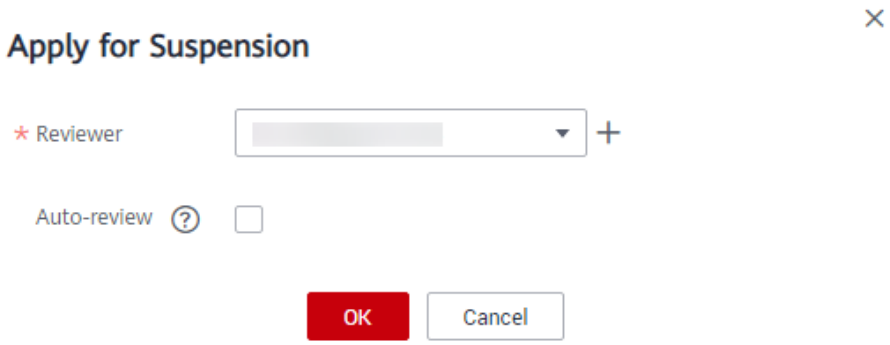
You can perform the following steps to suspend a published business metric:

Step 1 On the DataArts Architecture page, choose **Metrics** > **Business Metrics** in the left navigation pane.

Step 2 In the business metric list, select the target business metric and click **Suspend** in the **Operation** column.

Step 3 In the dialog box displayed, select a reviewer and click **OK**. The business metric is suspended after the reviewer approves it.

Figure 5-139 Apply for Suspension dialog box



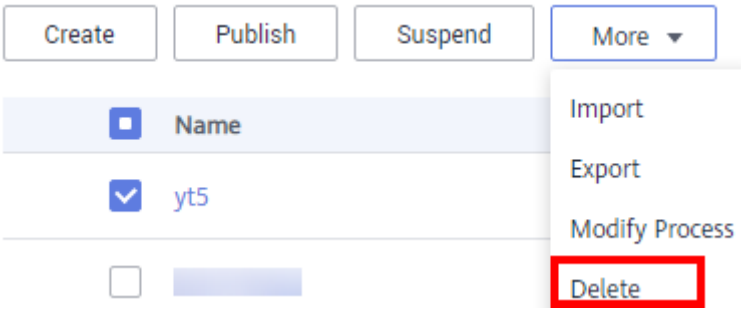
----End

Deleting a Business Metric

If a business metric is no longer needed, you can delete it. A business metric in the **Published** state can be deleted only after it is suspended.

1. On the DataArts Architecture page, choose **Metrics** > **Business Metrics** in the left navigation pane.
2. In the business metric list, select the target business metric and choose **More** > **Delete** above the list.

Figure 5-140 Deleting a business metric



3. In the dialog box displayed, confirm the information and click **Yes**.

Importing/Exporting Business Metrics

Importing metrics: You can import business metrics in batches.

1. On the DataArts Architecture page, choose **Metrics** > **Business Metrics** in the left navigation pane.
2. Above the business metric list, click **More** and select **Import**. In the displayed **Import Business Metric** dialog box, click **Business Metric Template**.

Figure 5-141 Importing business metrics

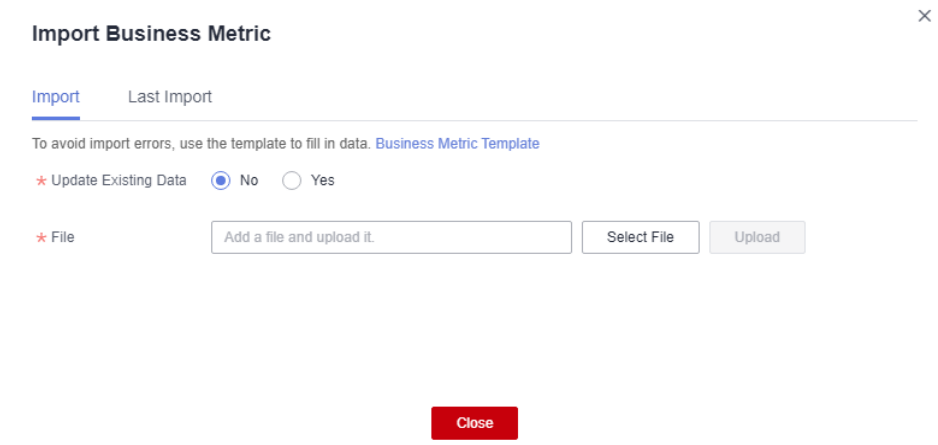


Table 5-45 Parameters for importing business metrics

Parameter	Description
Update Existing Data	Whether to update the existing table if the table to be imported already exists. The system determines whether the table to import exists based on the table code. The import operation can be used to create a table or update an existing table. It does not delete a table. The options are as follows: • **No**: If you select this option, the existing tables will not be updated. • **Yes**: If you select this option, the existing tables will be updated. If a table is in the **Published** state, you must publish the table again after updating it so that the updated table can take effect.
File	Select the file to import. You can use the following method to obtain the file to import: **Downloading the ER modeling template and fill in the template** On the **Import** tab page, click **Business Metric Template** to download the template, fill in the template, and save the settings.

3. Open the downloaded template, set the parameters in the template, and save the template. The **Description** sheet in the template is for reference only.

   Parameters whose names start with an asterisk (*) are mandatory, and other parameters are optional.

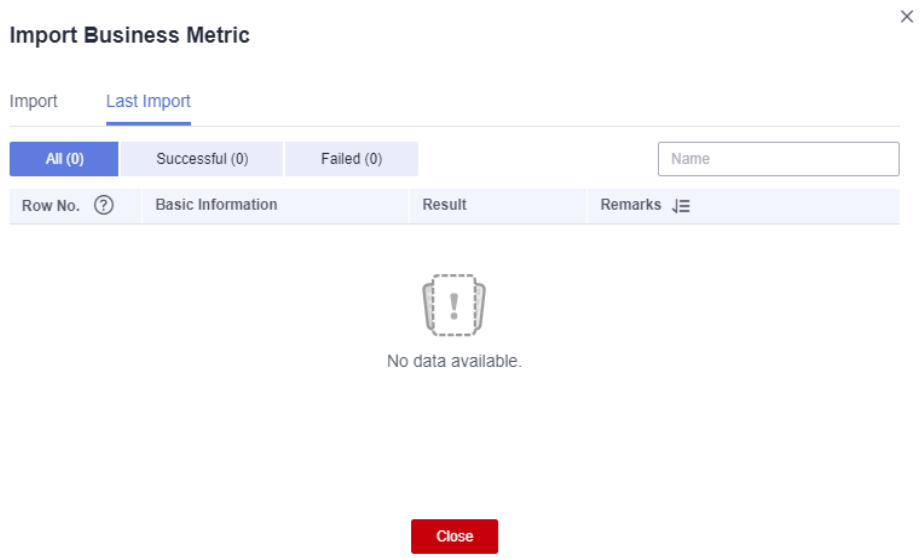
   The table below describes the parameters in the **Business Metric** sheet.

Table 5-46 parameters in the Business Metric sheet

Parameter	Description
*Process	Level-1 process corresponding to the metric
*Name	Standard name of the metric, which must be unique.
Code	It is automatically generated by the system.
Alias	Simplified name of the metric used in specific scenarios such as reports
*Objective	Objective of the metric
*Metric Definition	Accurate meaning of the metric that can help related personnel understand the content measured by the metric
*Calculation Formula	Clear rule for calculating the metric data
Data Source	System from which data comes. If possible, the specific data table name and field should be specified.
Measurement Unit	Basic measurement unit of the metric
*Statistical Period	Statistical period of the metric
Statistical Dimension	Common statistical dimension. Dimensions are generally hierarchical.
*Refresh Frequency	Minimum frequency at which the metric data is updated
Statistical Standard & Modifier	Statistical standard and modifier the metric usually uses in addition to the statistical period and dimension. The statistical standard and modifier restrict the scope of metric data.
Metric Application Scenario	Important application scenarios of the metric, such as online reports, routine reports, and reporting materials
Remarks	Additional information that helps understand and use the metric
Measurement Object	Field for measuring the metric. If this parameter is not involved, leave it blank.
*Metric Mgmt Dept	Department responsible for defining, maintaining, and interpreting the metric and providing metric data.
*Metric Owner	Metric owner (Huawei account name)
Related Tech Metric	Implementation of the business metric in the specifications design

4. View the result on the **Last Import** tab page. If the import is successful, click **Close**. If the import fails, you can view the failure cause, correct the template file, and upload it again.

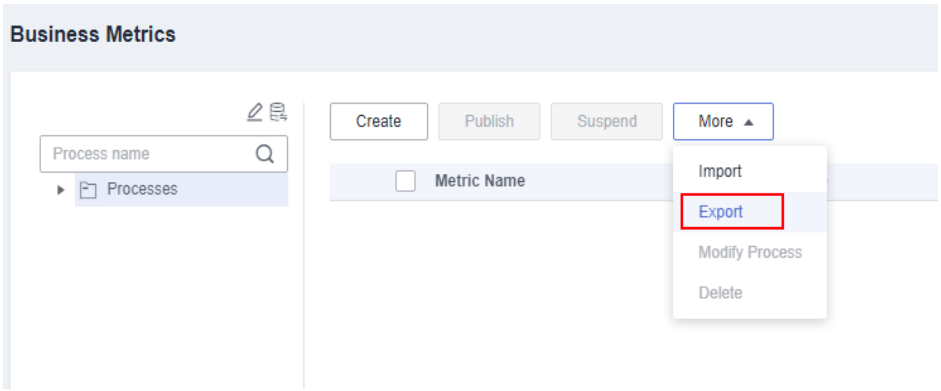
Figure 5-142 Last Import tab page



Exporting metrics: You can export created business metrics.

1. On the DataArts Architecture page, choose **Metrics** > **Business Metrics** in the left navigation pane.
2. On the **Business Metrics** page, select a business metric, click **More**, and select **Export**.

Figure 5-143 Exporting a business metric



NOTE

If a custom metric was created on the **Metrics** tab page in the configuration center, the custom metric is displayed in the exported table.

5.7.2 Technical Metrics

5.7.2.1 Creating Atomic Metrics

An atomic metric is an abstract set of the statistical logic and specific algorithms. To ensure consistency between definitions and R&D, metric definitions determine the statistical logic (or the computing logic), without using ETLs to perform secondary R&D. This improves R&D efficiency and ensures consistency of statistical results.

Atomic metrics are generated based on dimension tables and fact tables of a multidimensional model. The objects and the finest data granularity of an atomic metric are consistent with those of the multidimensional model.

An atomic metric usually consists of measures and attributes related with measures and business objects, all of which aim to support agile self-service consumption of the metric.

Context

Atomic metrics come from fact tables and dimension tables.

- An atomic metric is a data component defined for constructing a derivative metric required by application statistical analysis. An atomic metric can be created based on fact table details or dimension tables.
- A derivative metric does not have a direct source table. It belongs to the source table of the original atomic metrics that are combined into the derivative metric.

Atomic metrics and derivative metrics interact in specific ways.

- After the computing logic of an atomic metric takes effect, the related derivative metric is updated directly.
- An atomic metric referenced by any derivative metrics cannot be deleted.
- The code of an atomic metric referenced by any derivative metrics can be changed.
- The change of an atomic metric affects related derivative metrics.

Constraints

A maximum of 5,000 atomic metrics can be created in a workspace.

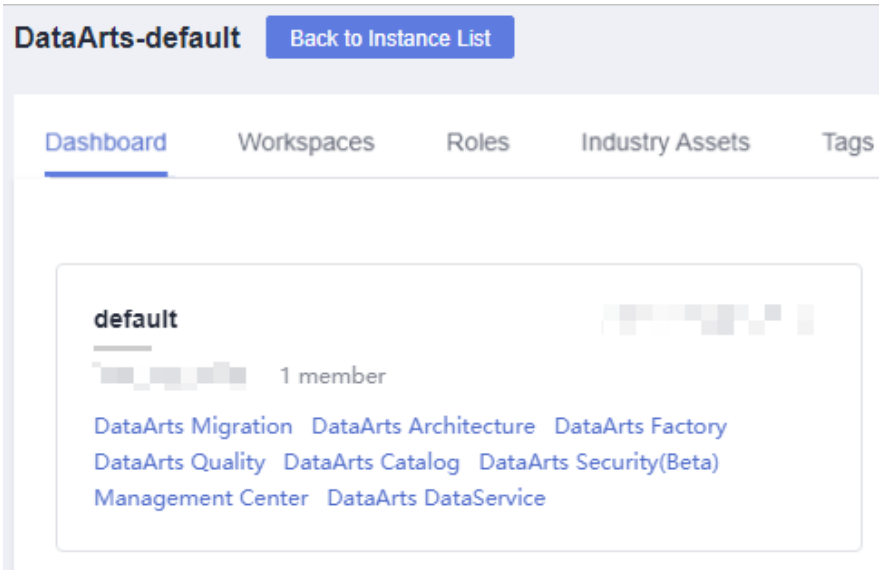
Prerequisites

You have created and published a fact table, and the fact table has been approved. For details, see **Creating Fact Tables**.

Creating and Publishing an Atomic Metric

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

Figure 5-144 DataArts Architecture



2. On the DataArts Architecture page, choose **Metrics** > **Technical Metrics** in the left navigation pane. On the displayed page, click the **Atomic Metrics** tab.
3. Select a subject from the subject tree on the left and click **Create**.
4. On the **Create Atomic Metric** page, set the parameters described in **Table 5-47** and click **Publish**.

Figure 5-145 Creating an atomic metric

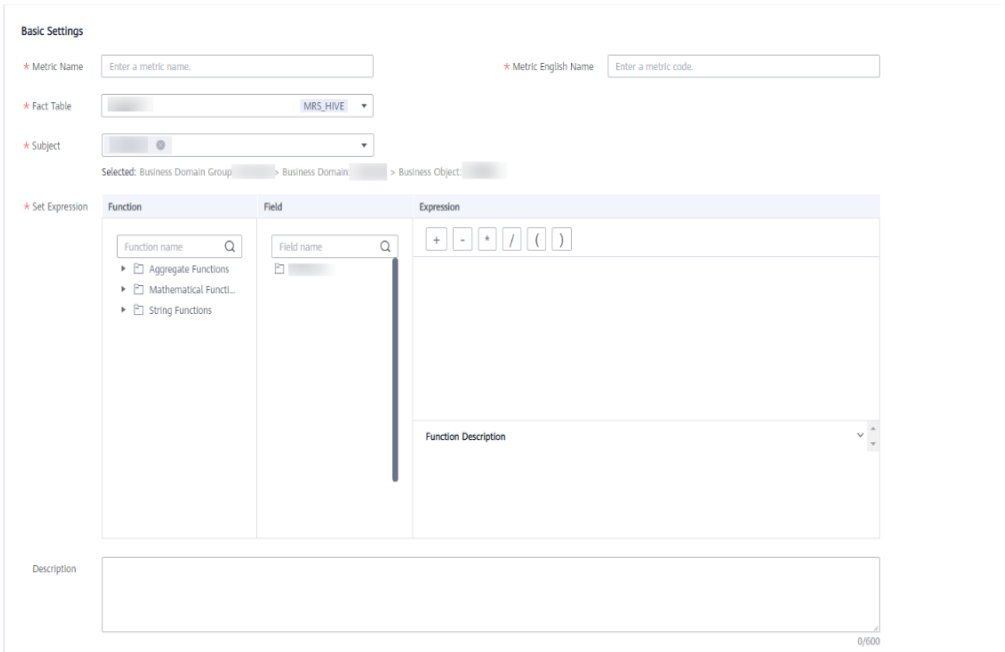


Table 5-47 Parameters for creating an atomic metric

Parameter	Description
*Metric Name	Metric names must start with letters. Only letters, numbers, and underscores (_) are allowed.
*Metric Code	Metric codes must start with letters. Only letters, numbers, and underscores (_) are allowed.
*Data Table	Select a published fact table from the drop-down list box. If there are many tables, you can enter a table name in the text box to search for the desired fact table. If no fact table is available, create one. See **Creating and Publishing a Fact Table**.
*Subject	The subject to which the atomic metric belongs. After a fact table is selected, the information about the subject to which the fact table belongs is automatically displayed. You can also click **Select** to select a subject.
*Set Expression	Select the required functions and fields and set the expression. For details about the functions, see **Functions**.
Description	A description of the atomic metric to create. Up to 600 characters are supported.

5. In the dialog box displayed, select a reviewer and click **OK**.
6. (Optional) Create and publish other atomic metrics by repeating **3** to **5**.
7. Wait for the reviewer to approve the application.

   After the application is approved, the atomic metric is created.

   Click the name of an atomic metric to view its details, relationship diagram, publishing history, and review history.

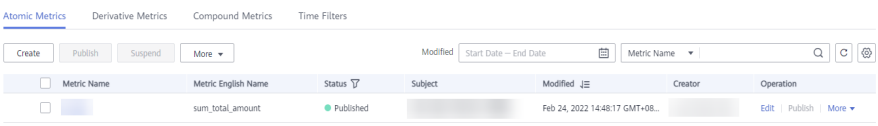
   In the relationship diagram, you can view the lineage diagram of the atomic metric.

   In the release history, you can view the differences between historical versions.

Managing an Atomic Metric

1. On the DataArts Architecture page, choose **Metrics** > **Technical Metrics** in the left navigation pane. On the displayed page, click the **Atomic Metrics** tab.

   Figure 5-146 Managing an atomic metric



2. Manage your atomic metrics as required. Refer to the following table for details.

Table 5-48 Operations

Operation	Helpful Link
Create	**Creating and Publishing an Atomic Metric**
Edit	**3**
Publish	**4**
View Publish History	**5**
Suspend	**6**
Delete	**7**
Import	**8**
Export	**9**

3. Edit an atomic metric.
   a. Click **Edit** to the right of the target atomic metric.
   b. On the page displayed, edit the atomic metric as required.
   c. Click **Publish**. If you do not want to immediately publish the atomic metric that you edited, click **Save** and you can publish it later.
4. Publish an atomic metric.
   a. Click **Publish** to the right of the target atomic metric.
   b. In the **Submit for Publication** dialog box displayed, select a reviewer from the drop-down list box.
   c. Click **OK**.
5. View the publish history.
   a. Select the target atomic metric in the list and choose **More** > **View History**.
   b. On the **History** tab page, you can view the publish history and version comparison information of the metric.
6. Suspend an atomic metric.
   a. Click **Suspend** to the right of the target atomic metric.
   b. In the **Submit for Suspension** dialog box displayed, select a reviewer from the drop-down list box.
   c. Click **OK**.

   NOTE

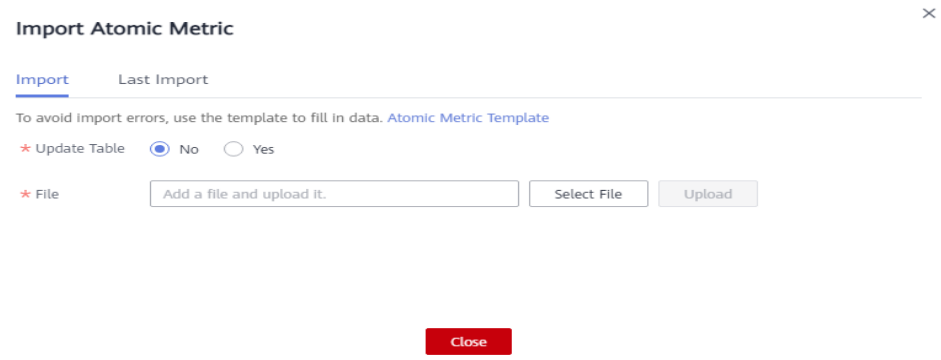
   Atomic metrics cannot be suspended or deleted if they are referenced by any derivative metrics.
7. Delete an atomic metric.
   a. Select the target atomic metric and choose **More** > **Delete** in the upper left corner.
   b. In the dialog box displayed, confirm the information and click **Yes**.

8. Import

   You can import atomic metrics to the system quickly.

   a. Above the atomic metric list, choose **More** > **Import**.

      Figure 5-147 Importing atomic metrics

   b. Download the atomic metric template, and edit and save it.

   c. Choose whether to update existing data.

      NOTE

      If a code in the template already exists in the system, the data is considered duplicate.

      - **No**: If the data to be imported already exists in the system, the existing data in the system will not be replaced.
      - **Yes**: If the data to be imported already exists in the system:
        - If the existing data in the system is in draft state, the data will be replaced and new draft data will be generated.
        - If the existing data in the system is in published state, expanded data will be generated.

   d. Click **Select File** and select the edited template to import.

   e. Click **Upload**. When the template is uploaded, the **Last Import** page is displayed. You can view the imported data.

   f. Click **Close**.

9. Export atomic metrics.

   You can export atomic metrics to a local file.

   a. In the atomic metric list, select the metric to be exported.

   b. Above the atomic metric list, choose **More** > **Export**.

      NOTE

      - You can export all the atomic metrics of a subject by selecting the subject in the subject list on the left.
      - You can export all the atomic metrics of a workspace, as long as there are no more than 5,000 atomic metrics in the workspace.

Functions

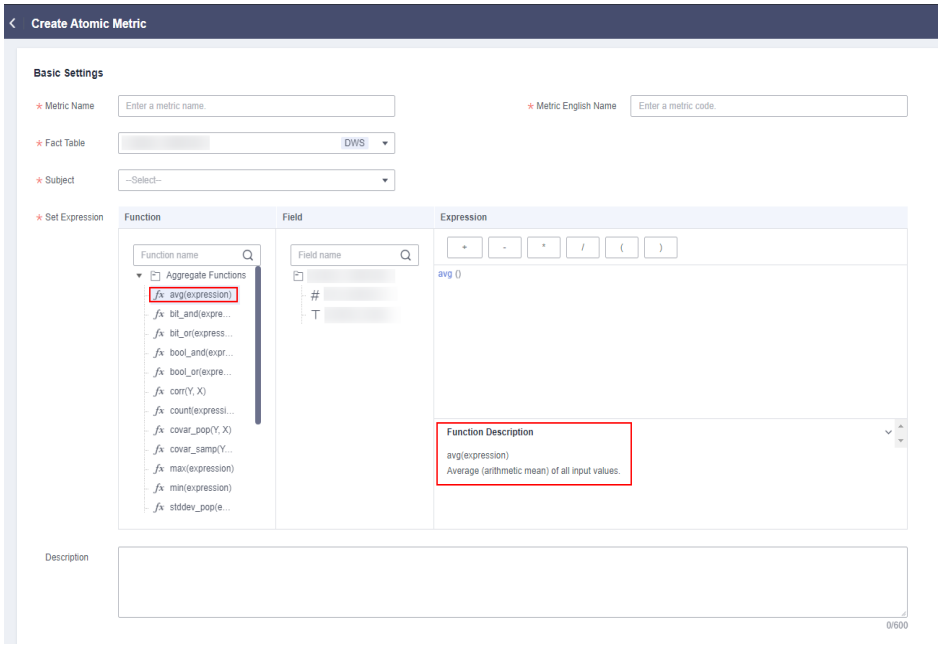
When creating an atomic metric, you need to set an expression based on functions. **Table 5-49** lists some aggregate functions.

Table 5-49 Aggregate functions

Function	Expression	Description
avg(col)	avg()	Returns the average value.
corr(col1, col2)	corr()	Returns the coefficient of correlation of a pair of numeric columns.
count(*)	count()	Returns the total number of records.
covar_pop(col1, col2)	covar_pop()	Returns the covariance of a pair of numeric columns.
covar_samp(col1, col2)	covar_samp()	Returns the sample covariance of a pair of numeric columns.
max(col)	max()	Returns the maximum value.
min(col)	min()	Returns the minimum value.
stddev_pop(col)	stddev_pop()	Returns the deviation of a specified column.
stddev_samp(col)	stddev_samp()	Returns the sample deviation of a specified column.
sum(col)	sum()	Returns the sum of the values in a column.
var_samp(col)	var_samp()	Returns the sample variance of a specified column.

You can click functions in the **Function** column next to **Set Expression** on the **Basic Settings** page on the **Create Atomic Metric** page.

Figure 5-148 Functions



5.7.2.2 Creating Derivative Metrics

Derivative metrics are aggregated from the modifiers and dimensions of atomic metrics. Therefore, their modifiers and dimensions are derived from the attributes of atomic metrics as well. When a derivative metric is published, a summary table is automatically generated, which can be viewed in the **Automatically Aggregated** area on the **Summary Table** tab page.

Derivative metric = Atomic metric + Dimension + Time filter + General filter

- **Atomic metric** specifies the statistical standards, namely, the computing logic.
- **Dimension** is the perspective to observe and analyze business data and assist in data aggregation, drilling, slicing, and analysis, and used as a GROUP BY condition in SQL statements.
- **Time filter** is a standard definition of a time condition.
- **General filter** collects statistics on the business scope and select the records that meet the business rules (similar to the WHERE clause in SQL statements, excluding the time range).

Prerequisites

- An atomic metric has been created and approved.
- A dimension and time filter have been created and approved. This prerequisite is required only if the derivative metric will use the statistical dimension or time filter.

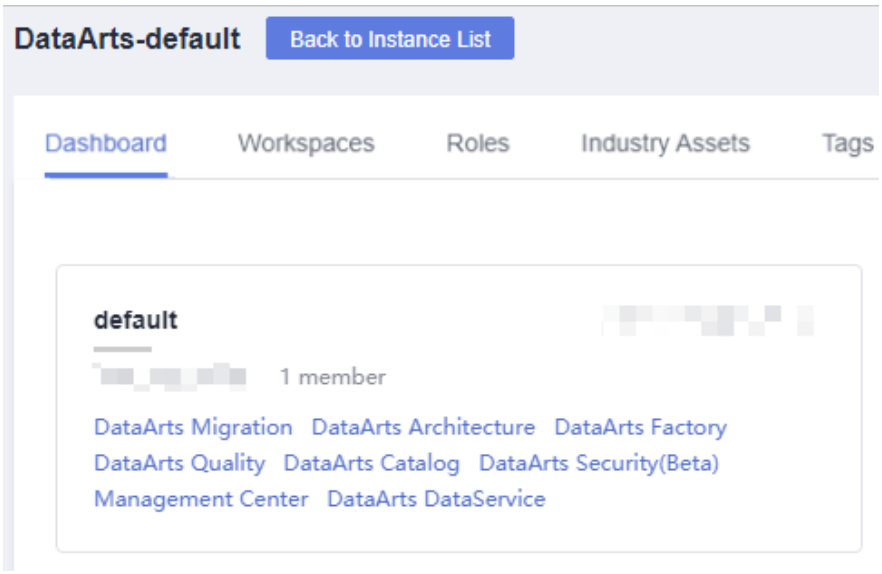
Constraints

A maximum of 5,000 derivative metrics can be created in a workspace.

Creating and Publishing a Derivative Metric

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

   Figure 5-149 DataArts Architecture

2. On the DataArts Architecture page, choose **Metrics** > **Technical Metrics** in the left navigation pane. On the displayed page, click the **Derivative Metrics** tab.
3. Select a subject from the subject tree on the left and click **Create**.
4. On the page displayed, set the parameters.

   Figure 5-150 Creating a derivative metric

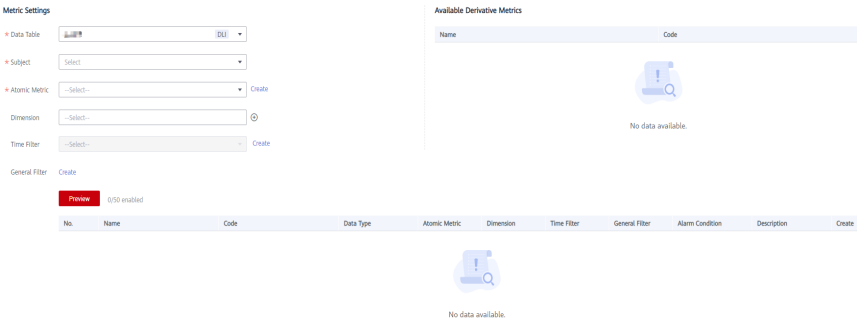
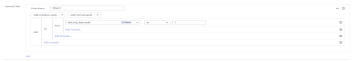
   

   Table 5-50 Parameters for creating a derivative metric

Parameter	Description
*Data Table	Select an asset table from the drop-down list box.

Parameter	Description
*Subject	Subject information.
*Atomic Metric	Select an atomic metric.
Dimension	Select one or more dimensions from the drop-down list box. Only the attributes in the fact table associated with the atomic metrics can be selected.
Time Filter	Select the required time filter from the drop-down list and select the associated field. Some time filters are preconfigured in the system. If the available time filters cannot meet the requirements, customize one. See **Creating Time Filters** for details.
General Filter	To set general filters, click **Create**. In the **General Filter** area shown in **Figure 5-151**, set the parameters as follows: • **Name** specifies the name of a general filter. • Under **Add Condition (and)**, you can select **And condition** or **Or condition** to add a condition. After you specify the condition, select a field from the field drop-down list and set the parameters as prompted. You can add multiple conditions. You can click to delete unwanted conditions. • Under **Add Formula (and)**, you can select **And formula** or **OR formula** to add a formula. Click **Edit Formula** if needed. In the dialog box displayed, select the required functions and fields and set the expression. You can click to delete unwanted formulas. **Figure 5-151** Setting a general filter 
Alarm Triggering Condition	An alarm triggering condition consists of derivative metrics and expressions. An expression consists of alarm parameters and logical operators. When a metric is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered.

5. After setting the parameters, click **Preview** to view the information about the derivative metric and define the name, code, data type, alarm condition, and description for the metric.

Table 5-51 Parameters for previewing a derivative metric

Parameter	Description
Metric Name	It is automatically generated by the system based on parameters such as atomic metrics, statistical dimensions, and time filters. You can also customize it.
Metric Code	It is automatically generated by the system based on parameters such as atomic metrics, statistical dimensions, and time filters. You can also customize it.
Data Type	It is automatically generated by the system based on the data type of the atomic metric. You can also customize it.
Alarm Condition	An alarm condition expression consists of alarm parameters and logical operators. When a metric is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered.
Description	A description of the derivative metric to create. Up to 600 characters are supported.

6. In the lower part of the page, click **Trial Run**. In the dialog box displayed, click **Trial Run** to check whether the derivative metric can run properly.

   If the trial run fails, locate the fault based on the error message, correct the configurations, and click **Trial Run** to try again.
7. If the trial run is successful, click **Publish**.
8. In the dialog box displayed, select a reviewer and click **OK**.
9. (Optional) Create and publish other derivative metrics by repeating **2** to **8**.
10. Wait for the reviewer to approve the application.

    After the application is approved, the derivative metric is created.

    Click the name of a derivative metric to view its details, relationship diagram, publishing history, and review history.

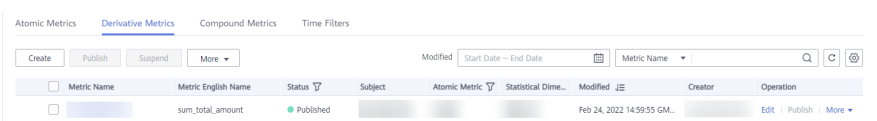
    In the relationship diagram, you can view the lineage diagram of the derivative metric.

    In the release history, you can view the differences between historical versions.

Managing a Derivative Metric

On the **Derivative Metrics** tab page, you can edit, publish, suspend, or delete derivative metrics.

Figure 5-152 Managing derivative metrics



1. On the DataArts Architecture page, choose **Metrics** > **Technical Metrics** in the left navigation pane. On the displayed page, click the **Derivative Metrics** tab.
2. Manage your derivative metrics as required. Refer to the following table for details.

Operation	Helpful Link
Create	**Creating and Publishing a Derivative Metric**
Edit	**3**
Publish	**4**
View Publish History	**5**
Preview SQL	**6**
Suspend	**7**
View Summary Table	**8**
Delete	**9**
Import	**10**
Export	**11**

3. Edit a derivative metric.
   a. Click **Edit** to the right of the target derivative metric.
   b. On the page displayed, edit the derivative metric as required.
   c. In the lower part of the page, click **Trial Run**. In the dialog box displayed, click **Trial Run** to check whether the derivative metric can run properly.

      If the trial run fails, locate the fault based on the error message, correct the configurations, and click **Trial Run** to try again.
   d. If the trial run is successful, click **Publish**.
4. Publish a derivative metric.
   a. Click **Publish** to the right of the target derivative metric.
   b. In the **Submit for Publication** dialog box displayed, select a reviewer from the drop-down list box.
   c. Click **OK**.
5. View the publish history.
   a. Select the target derivative metric and choose **More** > **View History**.
   b. On the page displayed, you can view the publish history and version comparison information of the metric.
6. Preview an SQL statement.
   a. Select the target derivative metric and choose **More** > **Preview SQL**.
   b. On the page displayed, you can view or copy the SQL statement.
7. Suspend a derivative metric.

NOTE

The prerequisite for suspending a derivative metric is that the derivative metric is not referenced to any compound metrics.

a. Choose **More** > **Suspend** on the right of the target derivative metric.
b. In the **Submit for Suspension** dialog box displayed, select a reviewer from the drop-down list box.
c. Click **OK**.

8. View a summary table.

   Currently, only details about automatically generated summary tables can be viewed. Choose **More** > **View Summary Table** on the right of the target derivative metric. The **Summary Tables** page is displayed.

9. Delete a derivative metric.

   NOTE

   The prerequisite for deleting a derivative metric is that the derivative metric is not referenced to any compound metrics.

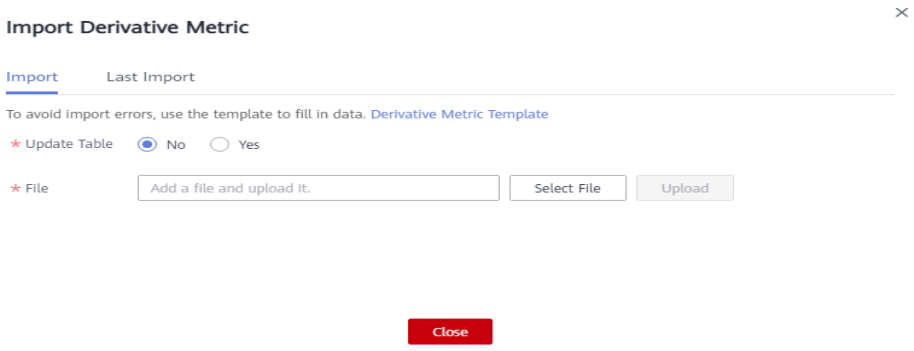
   a. Select the target derivative metric and choose **More** > **Delete** above the list.
   b. In the dialog box displayed, confirm the information and click **Yes**.

10. Import derivative metrics.

    You can import derivative metrics to the system quickly.

    a. Above the summary table list, choose **More** > **Import**.

    Figure 5-153 Importing derivative metrics

    b. Download the derivative metric template, and edit and save it.
    c. Choose whether to update existing data.

       NOTE

       If a code in the template already exists in the system, the data is considered duplicate.

       - **No**: If the data to be imported already exists in the system, the existing data in the system will not be replaced.
       - **Yes**: If the data to be imported already exists in the system:
         - If the existing data in the system is in draft state, the data will be replaced and new draft data will be generated.

- If the existing data in the system is in published state, expanded data will be generated.

d. Click **Select File** and select the edited template to import.

e. Click **Upload**. When the template is uploaded, the **Last Import** page is displayed. You can view the imported data.

f. Click **Close**.

11. Exporting derivative metrics.

    You can export derivative metrics to a local file.

    a. In the derivative metric list, select the metric to be exported.

    b. Above the derivative metric list, choose **More** > **Export**.

    NOTE

    - You can export all the derivative metrics of a subject by selecting the subject in the subject list on the left.
    - You can export all the derivative metrics of a workspace, as long as there are no more than 5,000 derivative metrics in the workspace.

5.7.2.3 Creating Compound Metrics

A compound metric is generated by adding one or more derivative metrics. The dimensions and modifiers of a compound metric are the same as those of the derivative metrics. New dimensions and modifiers cannot be generated outside the scope of derivative metrics, dimensions, and modifiers.

Constraints

A maximum of 5,000 compound metrics can be created in a workspace.

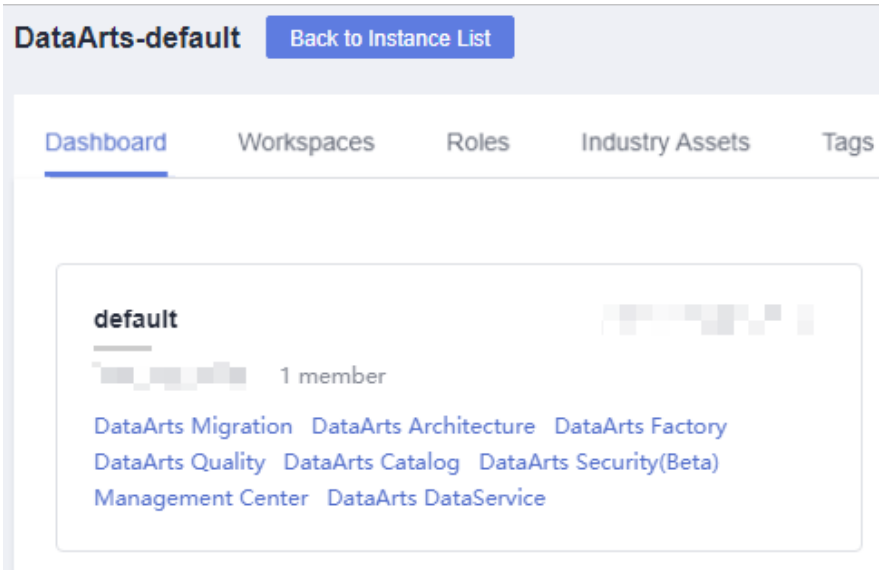
Prerequisites

A derivative metric has been created and approved. For details, see **Creating Derivative Metrics**.

Creating a Compound Metric

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

Figure 5-154 DataArts Architecture



2. On the DataArts Architecture page, choose **Metrics** > **Technical Metrics** in the left navigation pane. On the displayed page, click the **Compound Metrics** tab.
3. Select a subject from the subject tree on the left and click **Create**.
4. On the page displayed, set the parameters.

Figure 5-155 Creating a compound metric

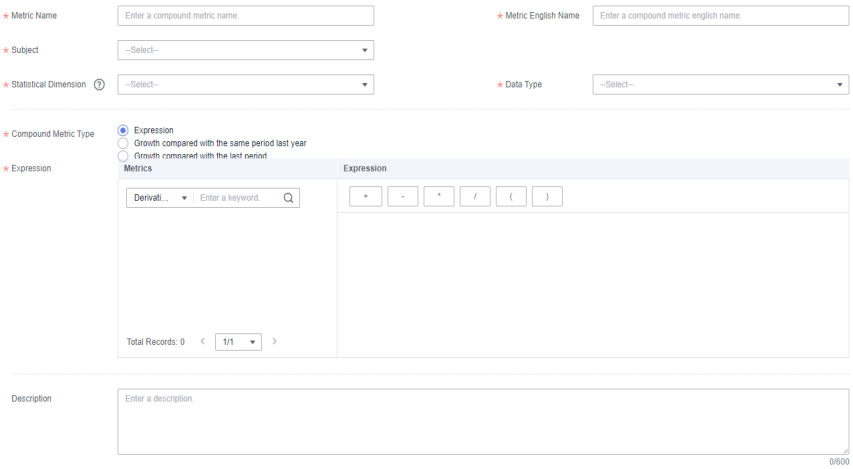


Table 5-52 Parameters for creating a compound metric

Parameter	Description
*Metric Name	Compound metric names must start with letters. Only letters, numbers, and underscores (_) are allowed.

Parameter	Description
*Metric Code	Metric code names must start with letters. Only letters, numbers, and underscores (_) are allowed.
*Subject	Subject information. Select a subject.
*Statistical Dimension	The available options are those configured on the **Derivative Metrics** page.
*Data Type	Select a data type for the compound metric.
*Compound Metric Type	The following options are available: • Expression • Growth compared with the same period last year • Growth compared with the last period
Description	A description of the compound metric to create. Up to 600 characters are supported.
Expression	
*Expression	Select the required derivative metrics or compound metrics and set the expression as required.
Growth compared with the same period last year	
*Period Type	Select **Year**, **Month**, or **Week**.
*Derivative Metric	Select derivative metrics (only derivative metrics with time filters are displayed). The system automatically calculates the growth compared with the same period last year based on the time filter.
Growth compared with the last period	
*Derivative Metric	Select derivative metrics (only derivative metrics with time filters are displayed). The system automatically calculates the growth compared with the last period based on the time filter.

5. In the lower part of the page, click **Trial Run**. In the dialog box displayed, click **Trial Run** to check whether the compound metric can run properly.

   If the trial run fails, locate the fault based on the error message, correct the configurations, and click **Trial Run** to try again.
6. If the trial run is successful, click **Publish**.
7. In the dialog box displayed, select a reviewer and click **OK**.
8. Wait for the reviewer to approve the application.

   After the application is approved, the compound metric is created.

   Click the name of a compound metric to view its details, relationship diagram, publishing history, and review history.

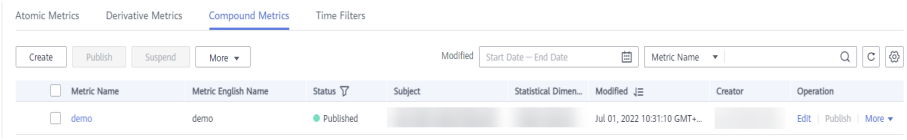
In the relationship diagram, you can view the lineage diagram of the compound metric.

In the release history, you can view the differences between historical versions.

Editing a Compound Metric

1. On the DataArts Architecture page, choose **Metrics** > **Technical Metrics** in the left navigation pane. On the displayed page, click the **Compound Metrics** tab.

   Figure 5-156 Compound metrics

2. In the compound metric list, select the target metric and click **Edit** on the right.
3. On the page displayed, set the parameters as prompted. For details, see **Table 5-52**.
4. In the lower part of the page, click **Trial Run**. In the dialog box displayed, click **Trial Run** to check whether the compound metric can run properly.

   If the trial run fails, locate the fault based on the error message, correct the configurations, and click **Trial Run** to try again.
5. If the trial run is successful, click **Publish**.
6. In the dialog box displayed, select a reviewer and click **OK**.

Publishing a Compound Metric

After creating or editing a compound metric, it takes effect only after it is published. Compound metrics in publishing review, published, or suspension review state cannot be published.

1. On the DataArts Architecture page, choose **Metrics** > **Technical Metrics** in the left navigation pane. On the displayed page, click the **Compound Metrics** tab.
2. In the compound metric list, select the target compound metric and click **Publish**.
3. In the dialog box displayed, click **OK**.

Viewing the Publish History

1. On the DataArts Architecture page, choose **Metrics** > **Technical Metrics** in the left navigation pane. On the displayed page, click the **Compound Metrics** tab.
2. Select the target compound metric in the list and choose **More** > **View History**.
3. On the page displayed, you can view the publish history and version comparison information of the metric.

Previewing an SQL Statement

1. On the DataArts Architecture page, choose **Metrics** > **Technical Metrics** in the left navigation pane. On the displayed page, click the **Compound Metrics** tab.
2. Locate the target compound metric and choose **More** > **Preview SQL**.
3. In the dialog box displayed, you can view or copy the SQL statement.

Suspending a Compound Metric

You can bring a published compound metric offline if it is no longer used.

NOTE

The prerequisite for suspending a compound metric is that the metric is not referenced to any summary table.

1. On the DataArts Architecture page, choose **Metrics** > **Technical Metrics** in the left navigation pane. On the displayed page, click the **Compound Metrics** tab.
2. In the compound metric list, select the target compound metrics and click **Suspend** above the list.
3. In the dialog box displayed, click **OK**.

Deleting a Compound Metric

NOTE

The prerequisite for deleting a compound metric is that the metric is not referenced to any summary table.

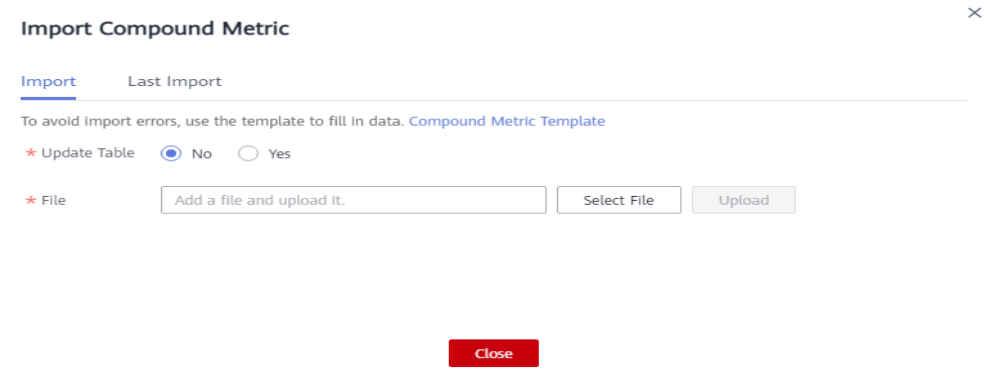
1. On the DataArts Architecture page, choose **Metrics** > **Technical Metrics** in the left navigation pane. On the displayed page, click the **Compound Metrics** tab.
2. Select the target compound metric and choose **More** > **Delete** above the list.
3. In the dialog box displayed, confirm the information and click **Yes**.

Importing Compound Metrics

You can import compound metrics to the system quickly.

1. Above the compound metric list, choose **More** > **Import**.

Figure 5-157 Importing Compound Metrics



2. Download the compound metric template, and edit and save it.
3. Choose whether to update existing data.

   NOTE

   If a code in the template already exists in the system, the data is considered duplicate.

   - **No**: If the data to be imported already exists in the system, the existing data in the system will not be replaced.
   - **Yes**: If the data to be imported already exists in the system:
     - If the existing data in the system is in draft state, the data will be replaced and new draft data will be generated.
     - If the existing data in the system is in published state, expanded data will be generated.
4. Click **Select File** and select the edited template to import.
5. Click **Upload**. When the template is uploaded, the **Last Import** page is displayed. You can view the imported data.
6. Click **Close**.

Exporting Compound Metrics

You can export compound metrics to a local file.

1. In the compound metric list, select the metric to be exported.
2. Above the compound metric list, choose **More** > **Export**.

NOTE

- You can export all the compound metrics of a subject by selecting the subject in the subject list on the left.
- You can export all the compound metrics of a workspace, as long as there are no more than 5,000 compound metrics in the workspace.

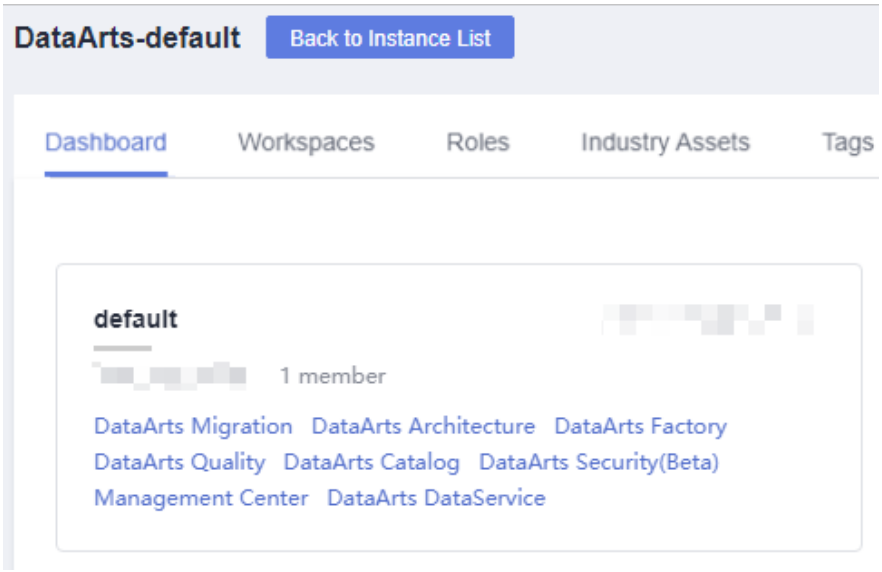
5.7.2.4 Creating Time Filters

Atomic metrics are standard definitions for computing logic. Time filters are standard definitions for conditional limits. To ensure that all statistical metrics are unified, standard, and unambiguous, time filters must be unique within a business domain and each filter can belong to a single source logic table. The computing logic is defined based on the fields of the source logic table model. A time filter may come from multiple logic tables that belong to different data domains. Therefore, a time filter may belong to multiple data domains as well.

Creating and Publishing a Time Filter

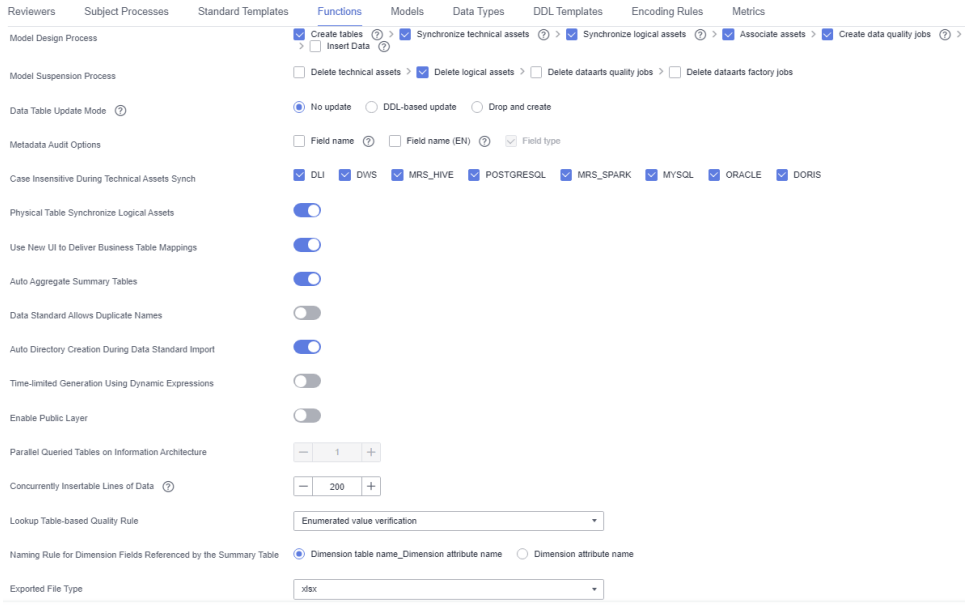
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

Figure 5-158 DataArts Architecture



2. (Optional) On the DataArts Architecture console, choose **Configuration Center** in the left navigation pane, click the **Functions** tab, and determine whether to enable **Time-Limited Generation Using Dynamic Expressions** (disabled by default).

Figure 5-159 Functions



3. On the DataArts Architecture page, choose **Metrics** > **Technical Metrics** in the left navigation pane. On the displayed page, click the **Time Filters** tab.
4. On the **Time Filters** tab page, click **Create**.
5. On the **Create Time Filter** page, set the parameters described in **Table 5-53** and click **Publish**.

Figure 5-160 Creating a time filter

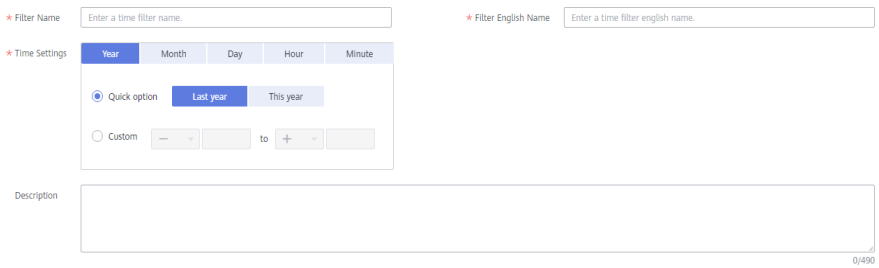


Table 5-53 Parameters for creating a time filter

Parameter	Description
*Filter Name	Time filter names must start with letters. Only letters, numbers, and underscores (_) are allowed.
*Filter English name	Only letters, digits, and underscores (_) are allowed.
*Time Settings	You can select **Year**, **Month**, **Day**, **Hour**, or **Minute**, and then select **Quick option** or **Custom** to set the time condition. If you select **Custom**, **+** and **-** form a time range, in which **+** indicates a later time and **-** indicates an earlier time. For example, if you want to set a time range from the past year to the next three years, set this parameter to **-1 to +3** or **+3 to -1**.
Description	A description of the time filter to create. Up to 490 characters are supported.

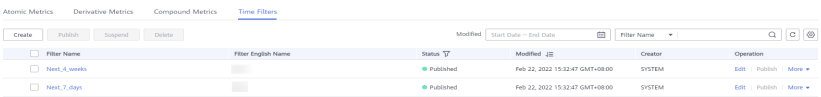
6. Then, click **Publish** to submit the application.
7. Wait for the reviewer to approve the application.

   After the application is approved, the time filter is created.

Managing a Time Filter

1. On the DataArts Architecture page, choose **Metrics** > **Technical Metrics** in the left navigation pane. On the displayed page, click the **Time Filters** tab.

   Figure 5-161 Time Filters tab page



2. Manage your time filters as required. Refer to the following table for details.

Operation	Helpful Link
Create	**Creating and Publishing a Time Filter**

Operation	Helpful Link
Edit	3
Publish	4
View Publish History	5
Suspend	6
Delete	7

3. Edit a time filter.
   a. Click **Edit** to the right of the target time filter.
   b. On the page displayed, edit the time filter as required.
   c. Click **Save** to save the time filter information, or click **Publish** to publish the edited time filter.
4. Publish a time filter.
   a. Click **Publish** to the right of the target time filter.
   b. In the **Submit for Publication** dialog box displayed, select a reviewer from the drop-down list box.
   c. Click **OK**.
5. View the publish history.
   a. Select the target time filter in the list and choose **More** > **View History**.
   b. On the page displayed, you can view the publish history and version comparison information of the time filter.
6. Suspend a time filter.
   a. Select the target time filter in the list and choose **More** > **Suspend**.
   b. In the **Submit for Suspension** dialog box displayed, select a reviewer from the drop-down list box.
   c. Click **OK**.

   NOTE

   Time filters cannot be suspended or deleted if they are referenced by any derivative metrics.
7. Delete a time filter.
   a. Select the target time filter and click **Delete** above the list.
   b. In the dialog box displayed, confirm the information and click **Yes**.

5.8 Data Mart Building

5.8.1 Creating Summary Tables

A summary table consists of specific analysis objects (such as members) and related statistical metrics. The metrics included in a summary table all have the

same level of granularity (such as members). A summary table provides users with all of the available statistics on themed data (such as a member theme market), sorted by levels of granularity.

A summary table can be manually or automatically aggregated. This topic describes how to manually create a summary table.

NOTE

On the DataArts Architecture page, choose **Metrics** > **Configuration Center** in the left navigation pane, and click the **Functions** tab. On the page displayed, if **Create data development jobs** is selected for **Model Design Process**, the system creates a data development job with a name starting with *Database name_Table code*. Choose **DataArts Factory** > **Develop Job** to view the created job. By default, this job has no scheduling configuration. You need to configure scheduling for the job in the DataArts Factory module.

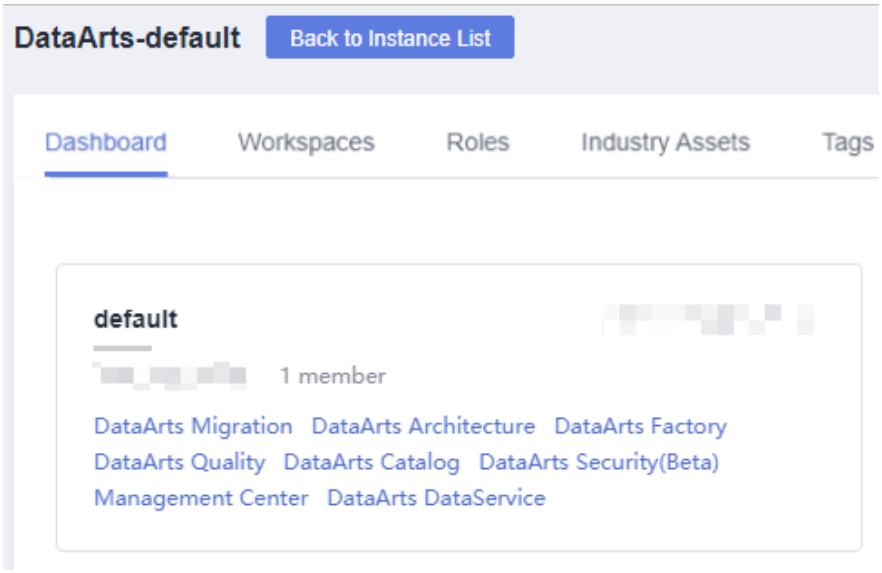
Prerequisites

A dimension, a dimension table, a fact table, and a derivative metric have been created, published, and reviewed.

Creating and Publishing a Summary Table

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

   Figure 5-162 DataArts Architecture

2. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane. On the displayed page, click the **Summary Tables** tab.
3. Select a subject from the subject tree on the left and click **Create**.
4. On the **Create Summary Table** page, perform the following operations:
   a. Set the parameters in the **Basic Settings** area.

Figure 5-163 Basic Settings area

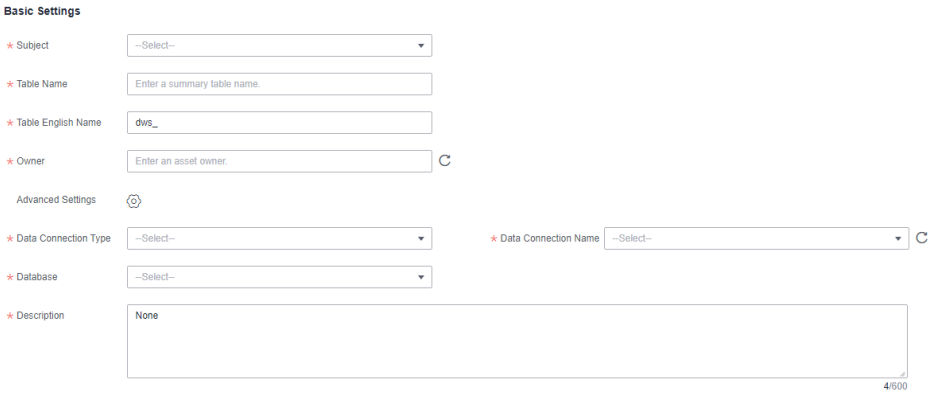


Table 5-54 Parameters in the Basic Settings area

Parameter	Description
*Subject	Select a subject catalog (business domain group > business domain > business object) where you can place the summary table.
*Table Name	The name of the table to create. It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_
*Table Code	The code of the table to create. Table codes must start with letters. Only letters, numbers, and underscores (_) are allowed.
*Owner	You can enter an owner name or select an existing owner.
Advanced Settings	Set custom items to describe the table. The custom items can be viewed in the table details. For example, if you want to identify the source of the table, you can add item **source** and set its value to the table source information. Then you can view the table source information in the table details.
*Data Connection Type	The parameter value must be the same as that of the dimension table and fact table.
*Data Connection Name	It is recommended that the same data connection be used for dimension modeling.
*Database	The name of the database. Select a database from the drop-down list box.
Queue	DLI queue. This parameter is available only for DLI data connections.

Parameter	Description
Schema	DWS or POSTGRESQL mode. This parameter is displayed only for DWS and POSTGRESQL data connections.
Table Type	DLI models support the following table types: • **MANAGED**: Data is stored in a DLI table. • **EXTERNAL**: Data is stored in an OBS table. When **Table Type** is set to **EXTERNAL**, you must set **OBS Path**. The OBS path format is ***/bucket_name/filepath***. DWS models support the following table types: • **DWS_ROW**: row-store table. Tables are stored to disk partitions by row. • **DWS_COLUMN**: column-store table. Tables are stored to disk partitions by column. • **DWS_VIEW**: view-store table. Tables are stored to disk partitions by view. The MRS Hive model supports **HIVE_TABLE** and **HIVE_EXTERNAL_TABLE**. The MRS Spark model supports **HUDI_COW** and **HUDI_MOR**. The PostgreSQL model supports only **POSTGRESQL_TABLE**. The MRS_CLICKHOUSE model supports only **CLICKHOUSE_TABLE**. The Oracle model supports only **ORACLE_TABLE**. The MySQL model supports only **MYSQL_TABLE**.
Compression Level	This parameter is available when the data connection type is DWS. The following compression levels are available for different table types: • **DWS_ROW**: **NO** and **YES** • **DWS_COLUMN**: **NO**, **LOW**, **MIDDLE**, and **HIGH**. • **DWS_VIEW**: The compression level is not supported.

Parameter	Description
Distributed By	This parameter is displayed only for DWS data connections. Currently, only **REPLICATION** and **HASH** are supported. You can select multiple fields. • **REPLICATION**: A full table is stored on each DN. The advantage of this option is that each DN has all the data of a table. During the join operation, data redistribution can be avoided, reducing network overhead. The disadvantage is that each DN retains the complete data of a table, resulting in data redundancy. Generally, this option is recommended for small dimension tables. • **HASH**: If you select this option, you must specify a distribution key for the user table. When a record is inserted, the system performs hash computing based on values in the distribute keys and then stores data on the corresponding DN. In a Hash table, I/O resources on each node can be used during data read/write, which improves the read/write speed of a table. Generally, this option is recommended for large dimension tables (a large dimension table contains over 1 million records).
* Description	A description of the summary table to create. It allows 1 to 600 characters.

b. Click the **Field Settings** tab and configure attributes for the summary table.

Click **Add** to add one or more associated attributes, for example, derivative metrics.

Click **Import Field** and select **From metrics**, **From dimension attributes**, or **Import from Data Standard**.

NOTE

If you select **From dimension attributes**, you must associate fields with metrics or import fields from metrics before associating fields with dimension attributes or importing fields from dimension attributes.

Click **Audit Data Standard** to audit the data standards of the attributes of the summary table. The audit status is ✓.

Click **Associate** to associate data standards or security levels with multiple attributes.

Click **Delete** to delete data standards or security levels from multiple attributes.

Figure 5-164 Configuring attributes

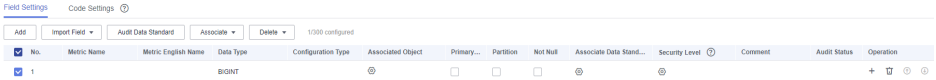


Table 5-55 Field parameters

Parameter	Description
Name	It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_ The surrogate key name of the added dimension field is displayed automatically. Generally, you do not need to change the name.
Name (EN)	It must start with letters. Only letters, numbers, and underscores (_) are allowed.
Data Type	Data type of the field name
Configuration Type	Configuration type corresponding to the field name, for example, derivative metric.
Associated Object	Associated object corresponding to the configuration type of the field name, for example, the derivative metric name
Primary Key	If this parameter is selected, the field is a primary key. **NOTE** If an MRS Spark connection is used to connect to MRS Hudi data sources, data can be written to the database only if fields have primary keys. Otherwise, table synchronization fails.
Partition	If this parameter is selected, the field is a partition field.
Not Null	Whether the parameter value can be left empty.
Data Standard	If you have created data standards, click  to select one to associate with the field. If **Create Data Quality Jobs** is selected for **Model Design Process** on the **Function Settings** tab page in Configuration Center and a field is associated with a data standard, a quality job is automatically generated after a table is published. A quality rule is generated for each field associated with the data standard. The quality of the field is monitored based on the data standard. You can access the **Quality Job** page of DataArts Quality to view the job details. If no data standard is available, create one. See **Creating Data Standards** for details.

Parameter	Description
Security Level	You can click to add a security level for the logical entity attribute. If you cannot find the security level you want, click **go to** to go to the DataArts Security console and create a security level. You can disable this function on the **Models** tab page on the **Configuration Center** page.
Description	Description
Audit Status	Whether to audit the data standard
Operation	Related operations

c. Click the **Code Settings** tab to view the code generated by the system and format the metric code.

You can click **Generate Code** to refresh the generated code, click **Copy to Metric Code** to copy the code to the metric code, and click **Format** to format the metric code.

5. Click **Publish**. In the dialog box displayed, click **OK**.

NOTE

You can choose to publish the summary table to the production or development environment. By default, it is published to the production environment. If you do not choose an environment, the summary table cannot be published.

6. Select a reviewer to approve the summary table.

After the summary table is approved, it is automatically created in the database.

7. Go back to the summary table list and locate the table just published. View its synchronization status in the **Sync Status** column. You can switch between the production environment and development environment to view the synchronization result.
   - If the synchronization is successful, the summary table is successfully published and created in the database.
   - If the synchronization failed, choose **More** > **View History** in the row where the summary table is located. On the page displayed, click the **History** tab to view logs. Troubleshoot the problem based on the logs. After the error is rectified, choose **More** > **Synchronize** above the summary table list to issue the synchronization command again. If the problem persists, contact technical support personnel.

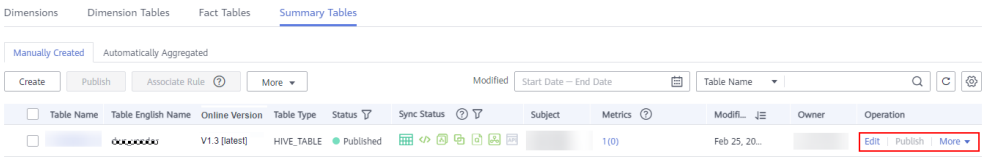
NOTE

You can choose to synchronize the summary table to the production or development environment. By default, it is synchronized to the production environment. If you do not choose an environment, the summary table cannot be synchronized.

Managing a Summary Table

1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane. On the displayed page, click the **Summary Tables** tab.

   Figure 5-165 Summary Tables page

2. Manage your summary tables as required. Refer to the following table for details.

Operation	Helpful Link
Create	Creating and Publishing a Summary Table
Edit	3
Publish	4
View History	5
Preview SQL	6
Suspend	7
Associate Rule	8
Delete	9
Import	10
Export	11

3. Edit a summary table.
   a. Click **Edit** to the right of the target summary table.
   b. Edit the summary table as required.
   c. Click **Publish**.

   NOTE

   You can choose to publish the summary table to the production or development environment. By default, it is published to the production environment. If you do not choose an environment, the summary table cannot be published.

4. Publish a summary table.
   a. Click **Publish** to the right of the target summary table.
   b. In the **Submit for Publication** dialog box displayed, select a reviewer from the drop-down list box.

      NOTE

      You can choose to publish the summary table to the production or development environment. By default, it is published to the production environment. If you do not choose an environment, the summary table cannot be published.

   c. Click **OK**.
5. View the publish history.
   a. Select the target summary table in the list and choose **More** > **View History** on the right.
   b. On the page displayed, you can view the publish history and version comparison information of the summary table.

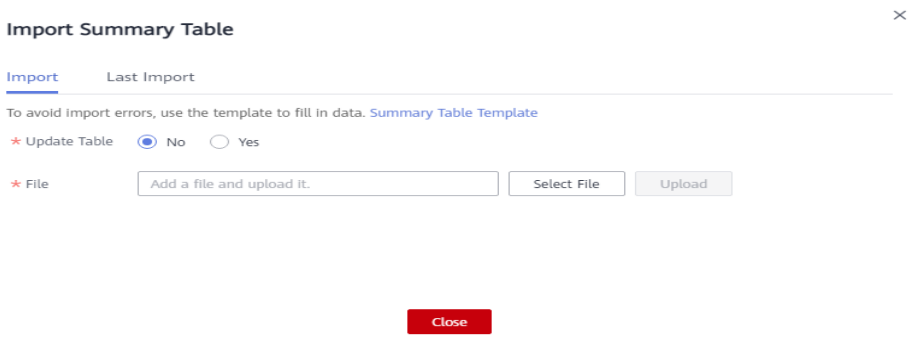
      If the publish log includes error logs, the publishing has failed. You can click **Resynchronize** to retry.
6. Previewing an SQL statement.
   a. Select the target summary table in the list and choose **More** > **Preview SQL** on the right.
   b. On the page displayed, you can view or copy the SQL statement.
7. Suspend a summary table.
   a. Click **Suspend** to the right of the target summary table.
   b. In the **Submit for Suspension** dialog box displayed, select a reviewer from the drop-down list box.
   c. Click **OK**.

      NOTE

      After a summary table is suspended, you can determine how to process APIs based on the actual situation in DataArts DataService. DataArts Architecture does not process the APIs.
8. Associate a summary table with a quality rule.
   a. Select the target summary table in the summary table list and click **Associate Rule** above the list.
   b. In the **Associate Quality Rule** dialog box, you can add rules to the fields in the summary table in batches and associate the rules with the fields.
   c. Click **OK**.
9. Delete a summary table.
   a. Select the target summary table and choose **More** > **Delete** above the list.
   b. In the dialog box displayed, click **Yes**.
10. Import

    You can import summary tables to the system quickly.
    a. Above the summary table list, choose **More** > **Import**.

Figure 5-166 Import Summary Table



b. Download the summary table template, and edit and save it.

c. Choose whether to update existing data.

NOTE

If a code in the template already exists in the system, the data is considered duplicate.

- **No**: If the data to be imported already exists in the system, the existing data in the system will not be replaced.
- **Yes**: If the data to be imported already exists in the system:
  - If the existing data in the system is in draft state, the data will be replaced and new draft data will be generated.
  - If the existing data in the system is in published state, expanded data will be generated.

d. Click **Select File** and select the edited template to import.

e. Click **Upload**. When the template is uploaded, the **Last Import** page is displayed. You can view the imported data.

f. Click **Close**.

11. Export summary tables.

You can export summary tables to a local file.

a. Select the summary tables to export on the **Manually Created** or **Automatically Aggregated** page.

b. Above the summary table list, choose **More** > **Export**.

NOTE

- You can export all the summary tables of a subject by selecting the subject in the subject list on the left.
- You can export all the summary tables of a workspace, as long as there are no more than 500 summary tables in the workspace.

Associating a Summary Table with a Quality Rule

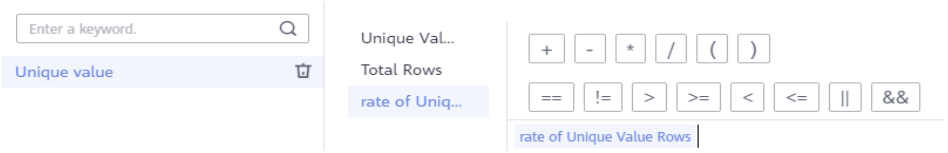
1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Summary Tables** tab.

3. Select the target summary table in the list, and click **Associate Rule**.

   Figure 5-167 Associating a summary table with a quality rule

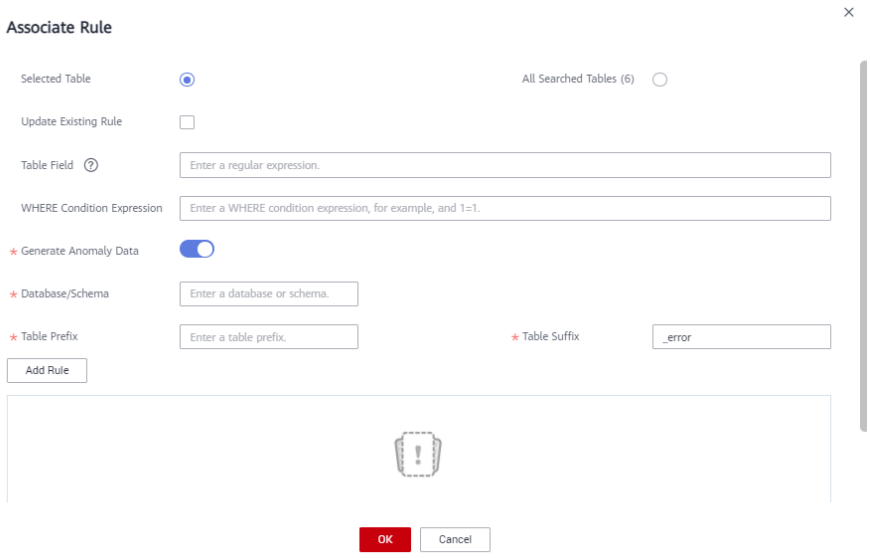
   

4. On the page displayed, set the parameters. After the configuration is complete, click **OK**.
   - **Update Existing Rule**: If this option is selected, the newly added rule will overwrite the old rule.
   - **Table Field**: This parameter applies to all fields by default. You can enter a regular expression to filter fields as required.
   - **WHERE Clause**: This parameter can be used to filter fields.
   - **Generate Anomaly Data**: If this option is selected, anomaly data is stored in the specified database based on the configured parameters.
   - **Database/Schema**: database or schema that stores anomaly data. This parameter is displayed when **Generate Anomaly Data** is enabled.
   - **Table Prefix**: prefix of the table that stores anomaly data. This parameter is displayed when **Generate Anomaly Data** is enabled.
   - **Table Suffix**: suffix of the table that stores anomaly data. This parameter is displayed when **Generate Anomaly Data** is enabled.
   - **Add Rule**: You can click **Add Rule** to add a rule. For example, add a rule named **Unique value**, select the rule, click **OK**, enter an alarm condition expression in the **Alarm Condition** text box, add other rules in the same way, and click **OK**. An example alarm expression is as follows:

   - An alarm condition expression consists of alarm parameters and logical operators. When a quality job is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered. In the **Associate Quality Rule** dialog box, the alarm parameters of each quality rule are displayed as buttons.

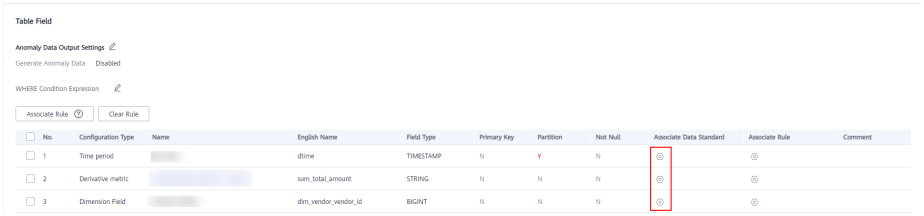
Figure 5-168 Associating a summary table with a quality rule



Associating a Summary Table Field with a Data Standard

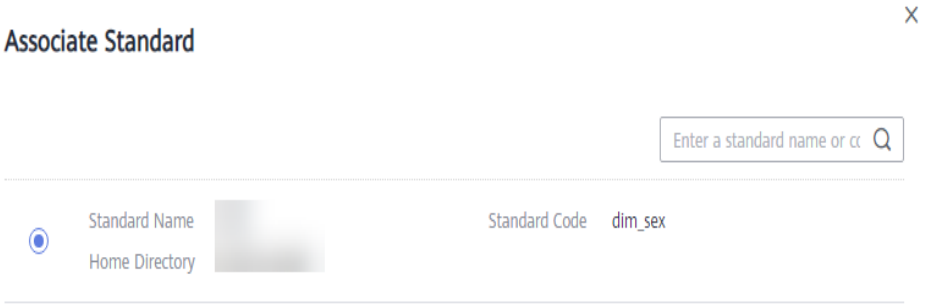
1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Summary Tables** tab.
3. Click the name of the target summary table in the list.
4. In the table field list on the details page of the summary table, search for the target field, click corresponding to the field to configure the association between the field and the data standard.

   Figure 5-169 Associating a summary table field with a data standard

5. After the configuration is complete, click **OK**. For details on the sources of data standards, see **Creating a Data Standard**.

Figure 5-170 Associating a data standard



Associating a Single Field with a Quality Rule

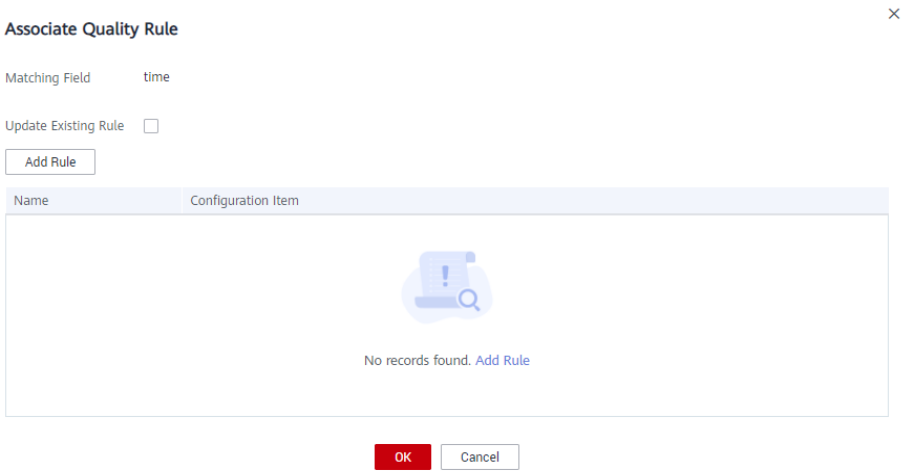
1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Summary Tables** tab.
3. In the summary table list, click the name of the target summary table.
4. In the table field list on the summary table details page, locate the target field and click to associate the field with a quality rule.

   Figure 5-171 Associating a single table field with a quality rule

5. After the configuration is complete, click **OK**.
   - **Update Existing Rule**: If this option is selected, the newly added rule will overwrite the old rule.
   - **Add Rule**: You can click **Add Rule** to add a rule. For example, add a rule named **Unique value**, select the rule, click **OK**, enter an alarm condition expression in the **Alarm Condition** text box, add other rules in the same way, and click **OK**.
   - An alarm condition expression consists of alarm parameters and logical operators. When a quality job is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered. In the **Associate Quality Rule** dialog box, the alarm parameters of each quality rule are displayed as buttons.

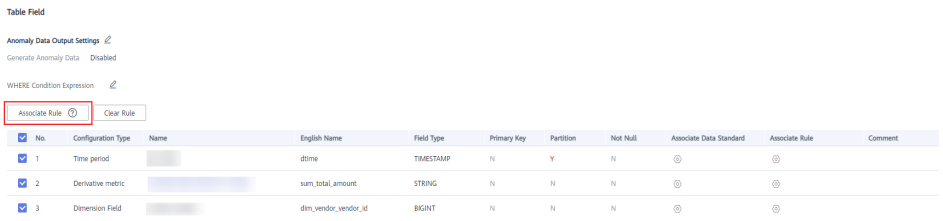
Figure 5-172 Associating a quality rule



Associating Table Fields with a Quality Rule in Batches

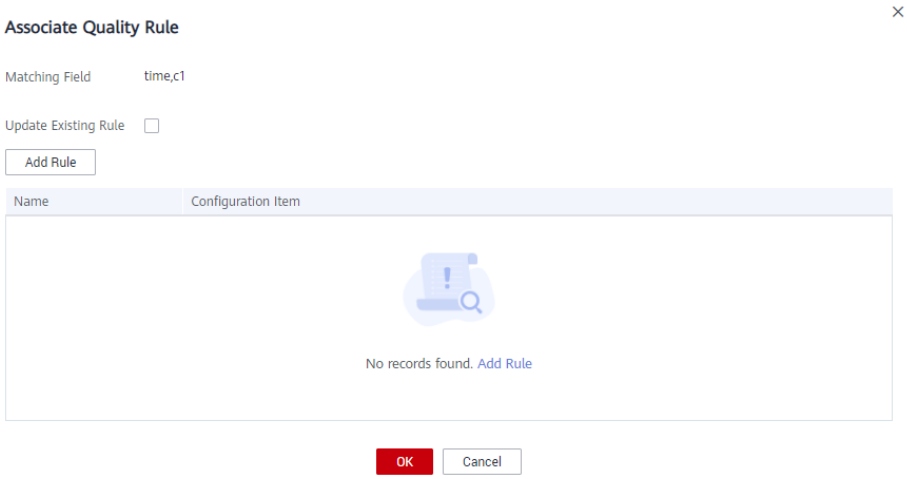
1. On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.
2. Click the **Summary Tables** tab.
3. In the summary table list, click the name of the target summary table.
4. In the table field list on the summary table details page, select the target table fields and click **Associate Rule**.

Figure 5-173 Associating fields with a quality rule



5. On the page displayed, add a rule and set the rule parameters.
   - **Update Existing Rule**: If this option is selected, the newly added rule will overwrite the old rule.
   - **Add Rule**: You can click **Add Rule** to add a rule. For example, add a rule named **Unique value**, select the rule, click **OK**, enter an alarm condition expression in the **Alarm Condition** text box, add other rules in the same way, and click **OK**.
   - An alarm condition expression consists of alarm parameters and logical operators. When a quality job is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered. In the **Associate Quality Rule** dialog box, the alarm parameters of each quality rule are displayed as buttons.

Figure 5-174 Adding a rule



6. After the configuration is complete, click **OK**.

5.9 Common Operations

5.9.1 Reversing a Database (ER Modeling)

You can import tables from databases of other data sources to a specific model.

Prerequisites

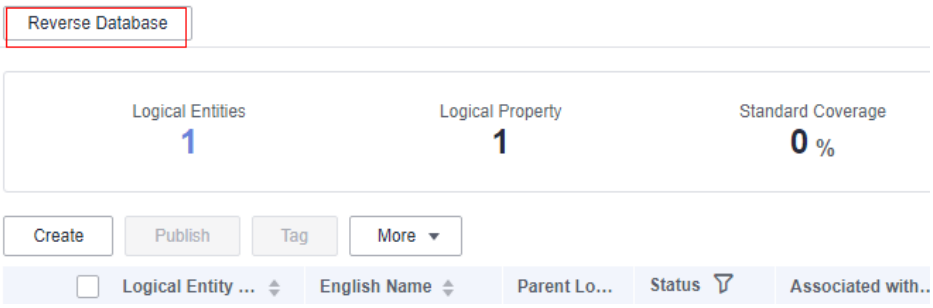
You have collected metadata from databases in DataArts Catalog so that the system can synchronize tables to DataArts Catalog later. Otherwise, the synchronization tasks may fail. See **Task Management** for details.

Importing a Table to a Model by Reversing the Database

Step 1 On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.

Step 2 On the model overview page, locate the target model, click the model card, and click **Reverse Database** in the upper part.

Figure 5-175 Reverse Database dialog box



Step 3 In the **Reverse Database** dialog box, set the parameters.

Figure 5-176 Setting parameters for reversing the database

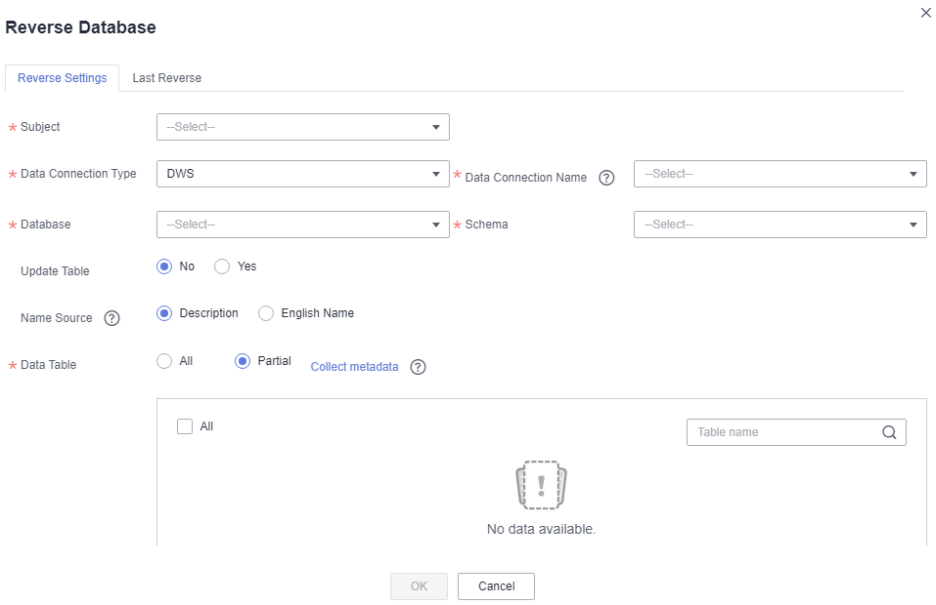


Table 5-56 Parameters for reversing a database

Parameter	Description
Subject	Select a subject from the drop-down list box.
Data Connection Type	If you reverse tables to a logical model, select a required data connection type from the drop-down list box. If you reverse tables to a physical model, the data connection type of the current model is displayed.
Data Connection	The name of the data connection. Select the required data connection. If you want to reverse a database from other data sources to an ER model, you must create a data connection in Management Center to connect to the data source. For details on how to create data connections, see **Managing Data Connections**.
Database	The name of the database. Select a database from the drop-down list box.
Queue	This parameter is displayed only for DLI data connections. Select a DLI queue.
Schema	Select a value from the drop-down list box. This parameter is available only for DWS and PostgreSQL tables.

Parameter	Description
Update Table	Whether to update the existing table if the table to be imported already exists in the ER model. When a table is imported, the system checks whether the table exists according to the table code. During the import, only table creation and update are allowed. • **No**: If you select this option, the existing tables will not be updated. • **Yes**: If you select this option, the existing tables will be updated. If a table is in the **Published** state, you must publish the table again after updating it so that the updated table can take effect.
Data Table	If you select **All**, all tables in the database are imported to the ER model. If you select **Partial**, not all tables in the database are imported to the ER model.
Start Page	This parameter is mandatory when **Data Table** is set to **All**.

Step 4 Click **Yes** to start reversing the database.

----End

5.9.2 Reversing a Database (Dimensional Modeling)

You can import tables from databases of other data sources to a specific model.

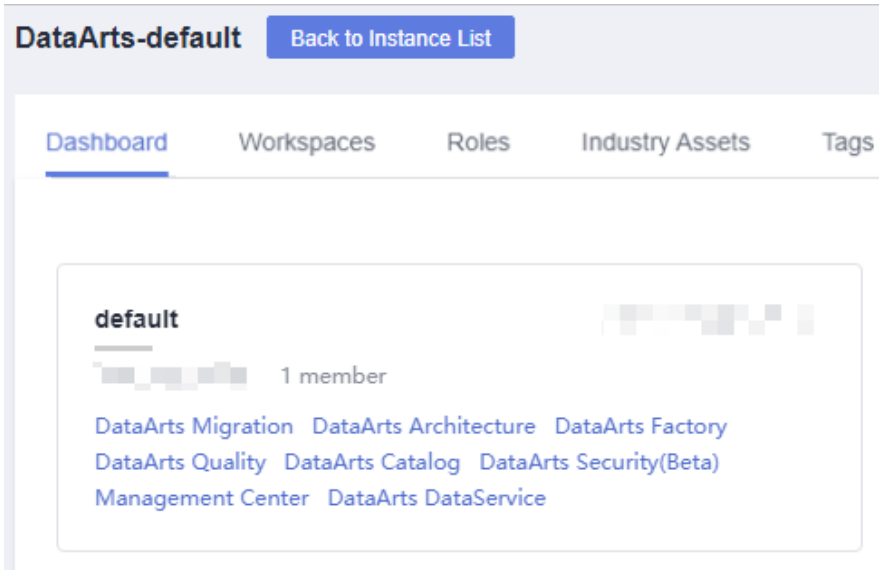
Prerequisites

You have collected metadata from databases in DataArts Catalog so that the system can synchronize tables to DataArts Catalog later. Otherwise, the synchronization tasks may fail. See **Task Management** for details.

Importing a Table to a Model by Reversing the Database

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

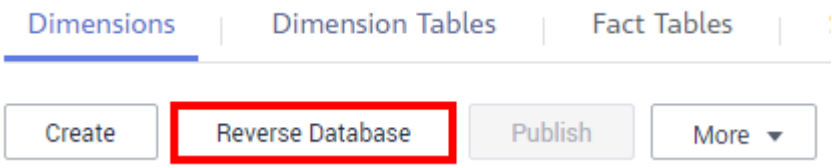
Figure 5-177 DataArts Architecture



Step 2 On the DataArts Architecture page, choose **Models** > **Dimensional Modeling** in the left navigation pane.

Step 3 Click the **Dimensions** or **Fact Tables** tab. Then, click **Reverse Database** above the list.

Figure 5-178 Selecting an object



Step 4 In the **Reverse Database** dialog box, set the parameters.

Table 5-57 Parameters for reversing a database

Parameter	Description
Subject	Select a subject from the drop-down list box.
Data Connection Type	Type of the database to reverse.
Data Connection	The name of the data connection. If you want to reverse a database from other data sources to an ER model, you must create a data connection in Management Center to connect to the data source. For details on how to create data connections, see **Managing Data Connections**.

Parameter	Description
Database	The name of the database. Select a database from the drop-down list box.
Queue	This parameter is displayed only for DLI data connections. Select a DLI queue.
Schema	DWS or POSTGRESQL mode. This parameter is displayed only for DWS and POSTGRESQL data connections.
Update Existing Table	The import operation can be used to create a table or update an existing table. It does not delete a table. • **No**: If you select this option, the existing tables will not be updated. • **Yes**: If you select this option, the existing tables will be updated. If a table is in the **Published** state, you must publish the table again after updating it so that the updated table can take effect.
Data Table	If you select **All**, all tables in the database are imported. If you select **Partial**, not all tables in the database are imported.

Step 5 Click **Yes** to start reversing the database. After the operation is complete, you can view the result on the **Last Reverse** tab page or perform the reverse operation again.

----End

5.9.3 Importing/Exporting Data

DataArts Architecture allows you to import and export processes, subjects, lookup tables, data standards, tables and entities in ER models, dimensions, fact tables, and summary tables in dimensional models, business metrics, and technical metrics. You cannot import or export time filters or data in the configuration and review centers.

This section describes how to import and export an ER modeling table. The operations for importing and exporting other data are similar. For details about how to import and export other data, see **DataArts Architecture Data Migration**.

Constraints

- Before importing tables/entities in ER modeling, and dimensions, fact tables, and summary tables in dimensional modeling, ensure that a data connection has been created in Management Center and is available.
- Time filters, and data in the Review Center and Configuration Center cannot be imported or exported. You must synchronize them manually before migrating other data.
- The maximum size of a file to be imported is 4 MB. A maximum of 3,000 metrics can be imported. A maximum of 500 tables can be exported at a time.

Importing a Table to a Logical Model

Step 1 On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.

Step 2 On the model overview page, click the card of the target logical model, select an object in the subject directory, and choose **More** > **Import**.

Step 3 In the dialog box displayed, click **ER Modeling Template**.

Figure 5-179 Import Table dialog box

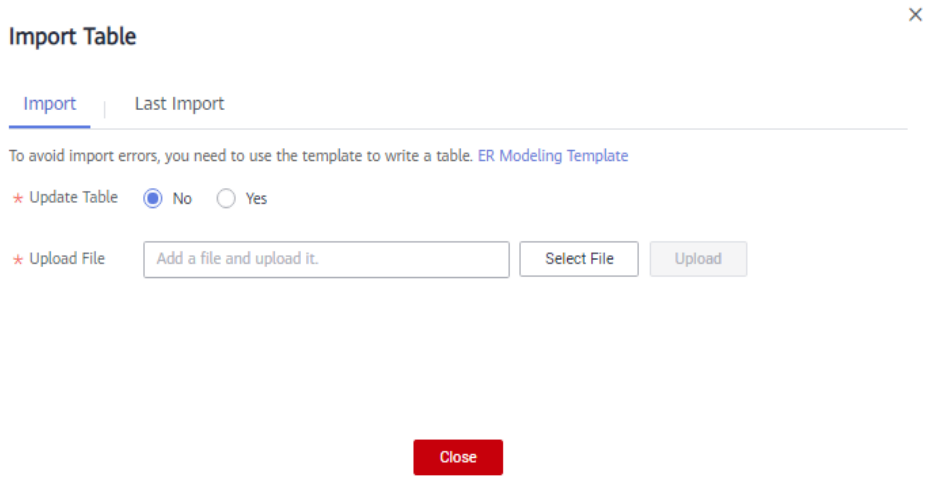


Table 5-58 Parameters for importing a table

Parameter	Description
Update Table	Whether to update the existing table if the table to be imported already exists in the ER model. The system determines whether the table to import exists in the ER model based on the table code. The import operation can be used to create a table or update an existing table. It does not delete a table. The options are as follows: • **No**: If you select this option, the existing tables will not be updated. • **Yes**: If you select this option, the existing tables will be updated. If a table is in the **Published** state, you must publish the table again after updating it so that the updated table can take effect.

Parameter	Description
Upload File	Select the file to import. You can use either of the following methods to obtain the file to import: • **Downloading the ER modeling template and fill in the template** In the **Import Table** dialog box, click **ER Modeling Template** to download the template, fill in the template, and save the settings. • **Exporting tables to files** You can export the lookup tables created in DataArts Architecture of a DataArts Studio instance to an Excel file. Then, import the Excel file. See **Exporting a Table or DDL** for details.

Step 4 Open the downloaded template, set the parameters in the template, and save the template. The **Description** sheet in the template is for reference only.

Parameters whose names start with an asterisk (*) are mandatory, and other parameters are optional.

The table below describes the parameters in the **Tables** sheet.

Table 5-59 Parameters in the Tables sheet

Parameter	Description
Subject	Enter the encoding paths of existing subjects, which are separated by slashes (/). If no subject is available, create one by referring to **Designing Subjects**.
*Logical Entity Name	It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_
*Table Name	Name of the table. Table codes cannot start with numbers. Only letters, numbers, and the following special characters are allowed: _${}
Table Alias	Alias of a table. This parameter is displayed when you have enabled **Table Alias** on the **Configuration Center** page.
Table Tags	Tags to be added to the table. Enter an existing or a new tag. You can add a tag on the **Tags** page of DataArts Catalog and go back to this page to select the tag. For details on how to add a tag, see **Tags**.
*Table Description	A description of the table.
Owner	You can enter an owner name or select an existing owner in the current workspace of the DataArts Studio instance.
Parent Table	You can enter only the names of other tables in this template.

Parameter	Description
DWS DISTRIBUTE BY	This field is required only for DWS data connections. The HASH (attribute name) and REPLICATION modes are supported.
*Field Name	The name of a field in the table. It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_
Field Code	The code of the field in the table. It must start with letters. Only letters, digits, and underscores (_) are allowed.
Code	Code of an attribute field in a table, which is automatically generated by the system
Field Alias	Alias of a field. This parameter is displayed when you have enabled **Field Alias** on the **Configuration Center** page.
Field Ordinal	Sequence number of the field in the table. The value starts from 1. This parameter is optional. If this parameter is left blank, fields are sorted in the sequence in the template by default.
Field Description	A description of the field.
*Field Data Type	Data type of the logical model. For details, see the DEFAULT group in **Field Types**.
Field Data Length	Data length. For a variable-length data type, specify the data length if a data connection type supports the data length. For example, for the DWS data connection type, if the field type is **CHAR(10)**, set **Field Data Type** to **CHAR** and **Field Data Length** to **10**.
Partition	The value **Y** indicates that the field is a partition field, and the value **N** indicates that the field is not a partition field.
Primary Key	The value **Y** indicates that the field is a primary key, and the value **N** indicates that the field is not a primary key.
Not Null	The value **Y** indicates that the field is not empty, and the value **N** indicates that the field can be empty.
Associate Data Standard	The code of the data standard to be associated. If no data standard is available, create one. See **Creating Data Standards** for details.
Field Tags	Tags to add to the field. Enter an existing tag or a new tag.. You can add a tag on the **Tags** page of DataArts Catalog and go back to this page to select the tag. For details on how to add a tag, see **Tags**.
configs	Enter the name and value of the custom item in **Advanced Settings**.

Step 5 The table below describes the parameters in the **Relations** sheet.

Table 5-60 Parameters in the Relations sheet

Parameter	Description
Relation Name	Name of the relation. It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_
*Child Table	Name of the child table in the relationship
*Child Table Field	Name of a field in the child table in the relationship. The field must be a foreign key of the child table and mapped to the primary key of the parent table.
*Child to Parent	Mapping of a child table to a parent table. The following values are available: • **1**: Each piece of data in the child table corresponds to only one piece of data in the parent table. • **0,1**: Each piece of data in the child table corresponds to at most one piece of data in the parent table. • **0..n**: One piece of data in the child table corresponds to multiple pieces of data in the parent table. • **1..n**: Each piece of data in the child table corresponds to at least one piece of data in the parent table.
*Parent to Child	Mapping of a parent table to a child table. The following values are available: • **1**: Each piece of data in the parent table corresponds to only one piece of data in the child table. • **0,1**: Each piece of data in the parent table corresponds to at most one piece of data in the child table. • **0..n**: One piece of data in the parent table corresponds to multiple pieces of data in the child table. • **1..n**: One piece of data in the parent table corresponds to at least one piece of data in the child table.
*Parent Table	Name of the parent table in the relationship
*Parent Table Field	Name of a field in the parent table in the relationship. The field must be a primary key of the parent table and mapped to the foreign key of the child table.
Role Name	Name of a custom role that identifies the relationship. It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_

Step 6 Enter the names of the associated tables and fields in the **Associated Rules** sheet.

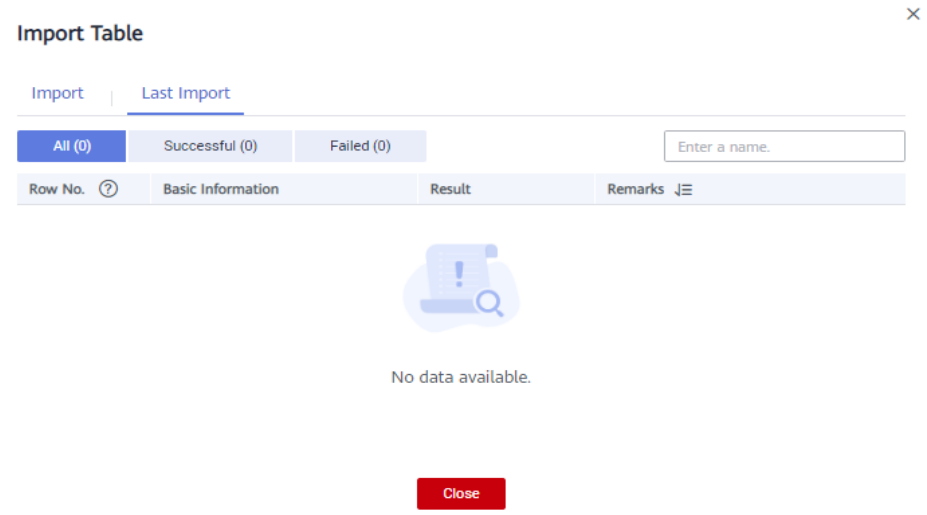
The table below describes the parameters in the **Associated Rules** sheet.

Table 5-61 Parameters in the Associated Rules sheet

Parameter	Description
*Table Name	Name of the table. It cannot start with digits. Only letters, digits, and the following special characters are allowed: _${}
*Field Name	The code of the field in the table. It must start with letters. Only letters, digits, and underscores (_) are allowed.
Rule Name	Enter an existing rule name. In the upper left corner of the DataArts Studio console, select **DataArts Quality** from the drop-down list box. Then, you can view the existing rule names on the **Rule Templates** page.
Alarm Triggering Condition	An alarm condition expression consists of alarm parameters and logical operators. When a quality job is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered. In the alarm condition expression, alarm parameters are represented by variables such as *${1}*, *${2}*, and *${3}*. The variable name indicates the alarm parameter of the specified quality rule. The variable *$1* indicates the first alarm parameter, *$2* indicates the second alarm parameter, and so on. In the upper left corner of the DataArts Studio console, select **DataArts Quality** from the drop-down list box. Access the **Rule Templates** page and view the alarm parameters supported by the data quality rule in the **Result Description** column. Example: ${1} > 100
Expression	An expression must be configured when **Rule Name** is set to **Expression** or **Validity Verification**.

Step 7 View the result on the **Last Import** tab page. If the import is successful, click **Close**. If the import fails, you can view the failure cause, correct the template file, and upload it again.

Figure 5-180 Last Import tab page



NOTE

- If the standard code associated with the imported logical entity does not exist or is not published, the system displays an error and the code name. In this case, change the code and try again.
- If the data to be imported does not exist, an error message in the format of *Table name*:*Field name* is displayed in the **Remarks** column on the **Last Import** tab page.

----End

Importing a Table to a Physical Model

Step 1 On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.

Step 2 In the ER model tree, select a physical model, expand it, and select a target. Then, choose **More** > **Import**.

Step 3 In the dialog box displayed, click **ER Modeling Template**.

Figure 5-181 Import Table dialog box

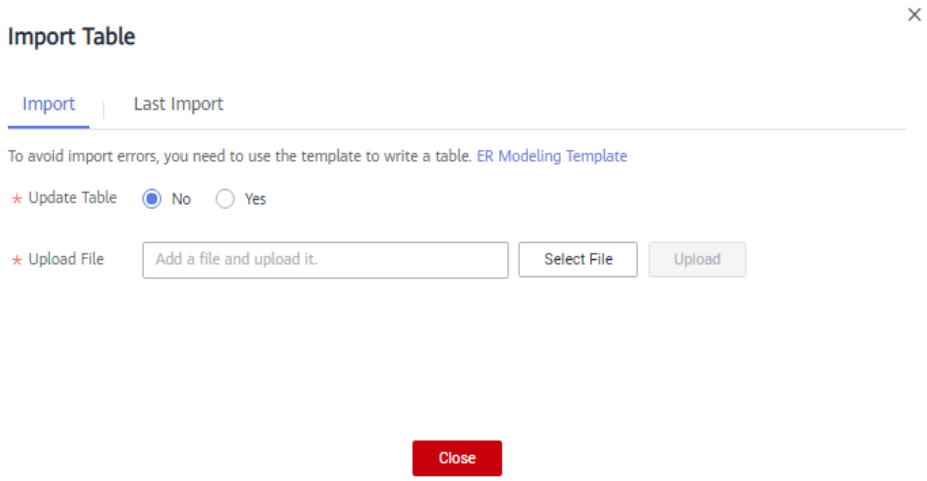


Table 5-62 Parameters for importing a table

Parameter	Description
Update Table	Whether to update the existing table if the table to be imported already exists in the ER model. The system determines whether the table to import exists in the ER model based on the table code. The import operation can be used to create a table or update an existing table. It does not delete a table. The options are as follows: • **No**: If you select this option, the existing tables will not be updated. • **Yes**: If you select this option, the existing tables will be updated. If a table is in the **Published** state, you must publish the table again after updating it so that the updated table can take effect.
Upload File	Select the file to import. You can use either of the following methods to obtain the file to import: • **Downloading the ER modeling template and fill in the template** In the **Import Table** dialog box, click **ER Modeling Template** to download the template, fill in the template, and save the settings. • **Exporting tables to files** You can export the lookup tables created in DataArts Architecture of a DataArts Studio instance to an Excel file. Then, import the Excel file. See **Exporting a Table or DDL** for details.

Step 4 Open the downloaded template, set the parameters in the template, and save the template. The **Description** sheet in the template is for reference only.

Parameters whose names start with an asterisk (*) are mandatory, and other parameters are optional.

The table below describes the parameters in the **Tables** sheet.

Table 5-63 Parameters in the Tables sheet

Parameter	Description (Importing DLI/ POSTGRESQL/DWS/MRS Hive Tables)
Subject	Enter the encoding paths of existing subjects, which are separated by slashes (/). If no subject is available, create one. For details, see **Designing Subjects**.
*Logical Entity Name	It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_
*Table Name	Name of the table. Table codes cannot start with numbers. Only letters, numbers, and the following special characters are allowed: _${}

Parameter	Description (Importing DLI/ POSTGRESQL/DWS/MRS Hive Tables)
Table Alias	Alias of a table. This parameter is displayed when you have enabled **Table Alias** on the **Configuration Center** page.
Table Tags	Tags to be added to the table. Enter an existing or a new tag. You can add a tag on the **Tags** page of DataArts Catalog and go back to this page to select the tag. For details on how to add a tag, see **Tags**.
*Table Description	A description of the table.
Asset Owner	Enter the username for entering the current workspace. Only the workspace admin, developer, or O&M personnel can be set as the designer.
Data Connection Type	The following connection types are supported: DWS, DLI, POSTGRESQL, and MRS Hive.
*Table Type	DLI models support the following table types: • **DLI_MANAGED**: Data is stored in a DLI table. • **DLI_EXTERNAL**: Data is stored in an OBS table. When **Table Type** is set to **DLI_EXTERNAL**, you must set **OBS Path**. • **DLI_VIEW** is available for import only. DWS models support the following table types: • DWS_ROW: row type • DWS_COLUMN: column type • DWS_VIEW: view type This parameter is unavailable for the tables created in MRS Hive models.
OBS Path	Enter an OBS path for storing the source data associated with the table if **Table Type** is set to **DLI_EXTERNAL**. The OBS path format is ***bucket_name/filepath***.
Data Format	This parameter is available only for tables created in DLI models. If the table type is **DLI_MANAGED**, the options of the data format are **Parquet** and **Carbon**. If the table type is **DLI_EXTERNAL**, the options of the data format are **Parquet**, **Carbon**, **CSV**, **ORC**, **JSON**, and **Avro**.
Data Connection	Enter the name of a created data connection.
Database	Enter the name of a created database.

Parameter	Description (Importing DLI/ POSTGRESQL/DWS/MRS Hive Tables)
Connection Extra	If **Data Connection Type** is **DLI**, enter a DLI queue name. If **Data Connection Type** is **DWS** or **POSTGRESQL**, enter a schema name.
DWS DISTRIBUTE BY	This field is required only for DWS data connections. The HASH (attribute name) and REPLICATION modes are supported.
HUDI PreCombineField	Version field. This field is mandatory only for the Hudi table.
*Field Name	The name of a field in the table. It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_
*Field Code	The code of the field in the table. It must start with letters. Only letters, digits, and underscores (_) are allowed.
Field Alias	Alias of a field. This parameter is displayed when you have enabled **Field Alias** on the **Configuration Center** page.
Field Ordinal	Sequence number of the field in the table. The value starts from 1. This parameter is optional. If this parameter is left blank, fields are sorted in the sequence in the template by default.
Field Description	A description of the field.
*Field Data Type	The supported data types vary depending on the data connection types. For details, see **Field Types**.
Field Data Length	For a variable-length data type, specify the data length if a data connection type supports the data length. For example, for the DWS data connection type, if the field type is **CHAR(10)**, set **Field Data Type** to **CHAR** and **Field Data Length** to **10**.
Partition	The value **Y** indicates that the field is a partition field, and the value **N** indicates that the field is not a partition field.
Primary Key	The value **Y** indicates that the field is a primary key, and the value **N** indicates that the field is not a primary key.
Not Null	The value **Y** indicates that the field is not empty, and the value **N** indicates that the field can be empty.

Parameter	Description (Importing DLI/ POSTGRESQL/DWS/MRS Hive Tables)
Associate Data Standard	The code of the data standard to be associated. This field can be left blank. If no data standard is available, create one. For details, see **Creating Data Standards**.
Field Tags	Tags to add to the field. Enter an existing tag or a new tag.. You can add a tag on the **Tags** page of DataArts Catalog and go back to this page to select the tag. For details on how to add a tag, see **Tags**.

configs	Additional table configuration details stored in JSON format. The format is as follows:
	{
	"option_name1": "value",
	"option_name2": "value"
	.....
	}
	Example:
	{
	"a1": "100",
	"a2": "30"
	}

Version	This parameter is optional.
configs	Enter the name and value of the custom item in **Advanced Settings**.

Step 5 The table below describes the parameters in the **Relations** sheet.

Table 5-64 Parameters in the Relations sheet

Parameter	Description
Relation Name	Name of the relation. It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_
*Child Table	Name of the child table in the relationship
Child Table Database	Name of the database to which the child table in the relationship belongs.
*Child Table Field	Name of a field in the child table in the relationship. The field must be a foreign key of the child table and mapped to the primary key of the parent table.

Parameter	Description
*Child to Parent	Mapping of a child table to a parent table. The following values are available: • **1**: Each piece of data in the child table corresponds to only one piece of data in the parent table. • **0,1**: Each piece of data in the child table corresponds to at most one piece of data in the parent table. • **0..n**: One piece of data in the child table corresponds to multiple pieces of data in the parent table. • **1..n**: Each piece of data in the child table corresponds to at least one piece of data in the parent table.
*Parent to Child	Mapping of a parent table to a child table. The following values are available: • **1**: Each piece of data in the parent table corresponds to only one piece of data in the child table. • **0,1**: Each piece of data in the parent table corresponds to at most one piece of data in the child table. • **0..n**: One piece of data in the parent table corresponds to multiple pieces of data in the child table. • **1..n**: One piece of data in the parent table corresponds to at least one piece of data in the child table.
*Parent Table	Name of the parent table in the relationship
Parent Table Database	Name of the database to which the parent table in the relationship belongs.
*Parent Table Field	Name of a field in the parent table in the relationship. The field must be a primary key of the parent table and mapped to the foreign key of the child table.
Role Name	Name of a custom role that identifies the relationship. It must start with letters. Only letters, digits, and the following special characters are allowed: ()-_

Step 6 Enter the names of the associated tables and fields in the **Associated Rules** sheet.

The table below describes the parameters in the **Associated Rules** sheet.

Table 5-65 Parameters in the Associated Rules sheet

Parameter	Description
*Table Name	Name of the table. It cannot start with digits. Only letters, digits, and the following special characters are allowed: _${}
*Field Name	The code of the field in the table. It must start with letters. Only letters, digits, and underscores (_) are allowed.

Parameter	Description
Rule Name	Enter an existing rule name. In the upper left corner of the DataArts Studio console, select **DataArts Quality** from the drop-down list box. Then, you can view the existing rule names on the **Rule Templates** page.
Alarm Triggering Condition	An alarm condition expression consists of alarm parameters and logical operators. When a quality job is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered. In the alarm condition expression, alarm parameters are represented by variables such as *${1}*, *${2}*, and *${3}*. The variable name indicates the alarm parameter of the specified quality rule. The variable *$1* indicates the first alarm parameter, *$2* indicates the second alarm parameter, and so on. In the upper left corner of the DataArts Studio console, select **DataArts Quality** from the drop-down list box. Access the **Rule Templates** page and view the alarm parameters supported by the data quality rule in the **Result Description** column. Example: ${1} > 100
Expression	An expression must be configured when **Rule Name** is set to **Expression** or **Validity Verification**.

Step 7 View the import result on the **Last Import** tab page. If the import is successful, click **Close**. If the import fails, you can view the failure cause, correct the template file, and upload it again.

NOTE

- If the standard code associated with the imported logical entity does not exist or is not published, the system displays an error and the code name. In this case, change the code and try again.
- If the data to be imported does not exist, an error message in the format of *Table name*:*Field name* is displayed in the **Remarks** column on the **Last Import** tab page.

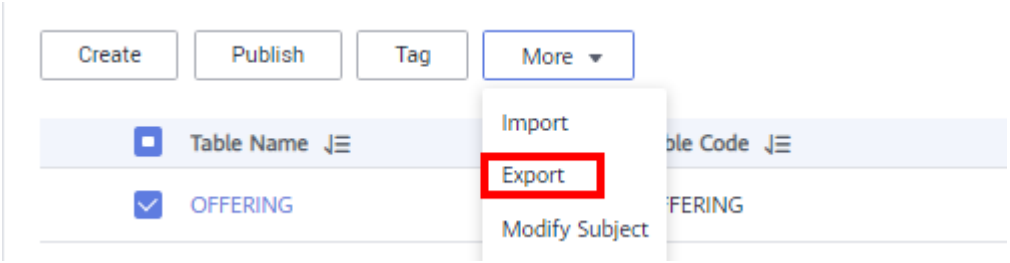
----End

Exporting a Table or DDL

Step 1 On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.

Step 2 On the model overview page, click the card of the target logical model, select an object in the subject directory, and choose **More** > **Export**.

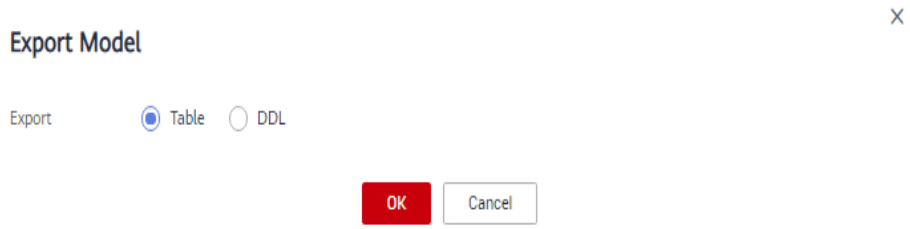
Figure 5-182 Exporting a table or DDL



Step 3 In the dialog box displayed, select the objects to export.

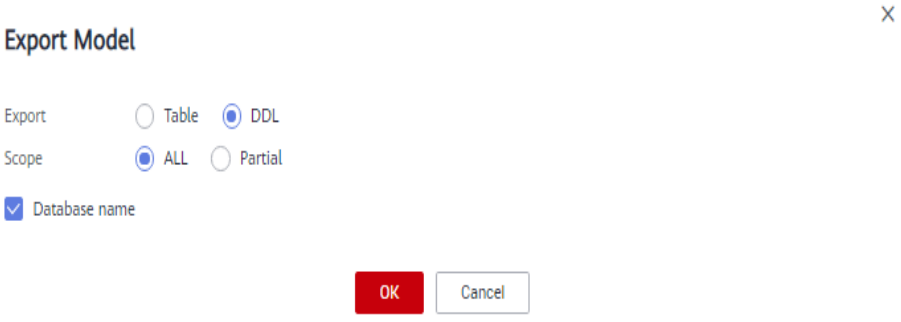
The exported Excel file can be imported.

Figure 5-183 Exporting a table



When a DDL is exported, the DDL statements of the selected table are exported to TXT files.

Figure 5-184 Exporting a DDL



Step 4 Click **OK**.

----End

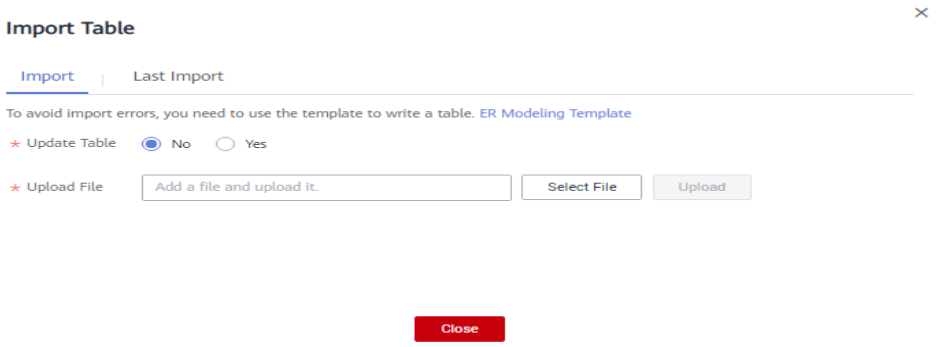
Importing/Exporting Dimensions

- **Importing dimensions**

  You can import dimensions to the system quickly.

  a. Above the dimension list, choose **More** > **Import**.

Figure 5-185 Import Table dialog box



b. Download the dimension template, and edit and save it.

c. Choose whether to update existing data.

NOTE

If a code in the template already exists in the system, the data is considered duplicate.

- **No**: If the data to be imported already exists in the system, the existing data in the system will not be replaced.
- **Yes**: If the data to be imported already exists in the system:
  - If the existing data in the system is in draft state, the data will be replaced and new draft data will be generated.
  - If the existing data in the system is in published state, expanded data will be generated.

d. Click **Select File** and select the edited template to import.

e. Click **Upload**. When the template is uploaded, the **Last Import** page is displayed. You can view the imported data.

f. Click **Close**.

NOTE

If the standard code associated with the imported logical entity does not exist or is not published, the system displays an error and the code name. In this case, change the code and try again.

- **Exporting dimensions**

  You can export dimensions to a local file.

  Above the dimension list, choose **More** > **Export**.

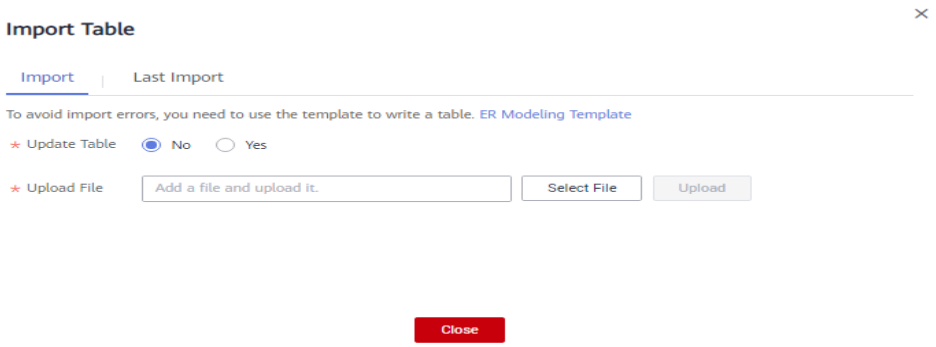
Importing/Exporting Fact Tables

- **Importing fact tables**

  You can import fact tables to the system quickly.

  a. Above the fact table list, choose **More** > **Import**.

Figure 5-186 Import Table dialog box



b. Download the fact table template, and edit and save it.

c. Choose whether to update existing data.

NOTE

If a code in the template already exists in the system, the data is considered duplicate.

- **No**: If the data to be imported already exists in the system, the existing data in the system will not be replaced.
- **Yes**: If the data to be imported already exists in the system:
  - If the existing data in the system is in draft state, the data will be replaced and new draft data will be generated.
  - If the existing data in the system is in published state, expanded data will be generated.

d. Click **Select File** and select the edited template to import.

e. Click **Upload**. When the template is uploaded, the **Last Import** page is displayed. You can view the imported data.

f. Click **Close**.

NOTE

If the standard code associated with the imported logical entity does not exist or is not published, the system displays an error and the code name. In this case, change the code and try again.

- **Exporting fact tables**

  You can export fact tables to a local file.

  Above the fact table list, choose **More** > **Export**.

5.9.4 Associating Quality Rules

After creating and publishing a table, you can associate quality rules with the table. If **Create Data Quality Jobs** is selected for **Model Design Process** on the **Function Settings** tab page of **Configuration Center**, a quality job is automatically created in DataArts Quality after a quality rule is associated and the table is published. If the table has been published, the system automatically updates the corresponding quality job.

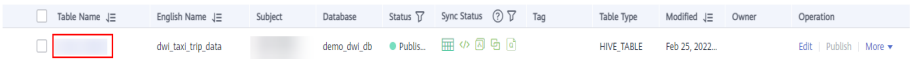
Associating a Quality Rule and Viewing a Quality Job

Step 1 On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.

Step 2 On the page displayed, select the target model. All tables created in the model are listed on the right. You can also expand a topic structure and select an object. All tables of the object are listed on the right.

Step 3 In the table list, select a table and click its name to access the table details page.

Figure 5-187 ER model list



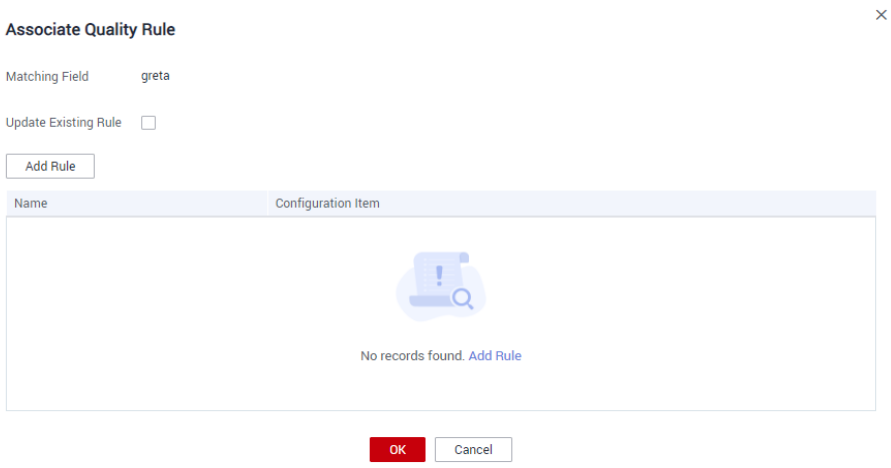
Step 4 In the **Table Field** area, select a field that you want to associate a quality rule with and click **Associate Rule**.

Figure 5-188 Associating Quality Rules

Anomaly Data Output Settings: If you select **Generate Anomaly Data**, the anomaly data is stored in the specified database based on the settings.

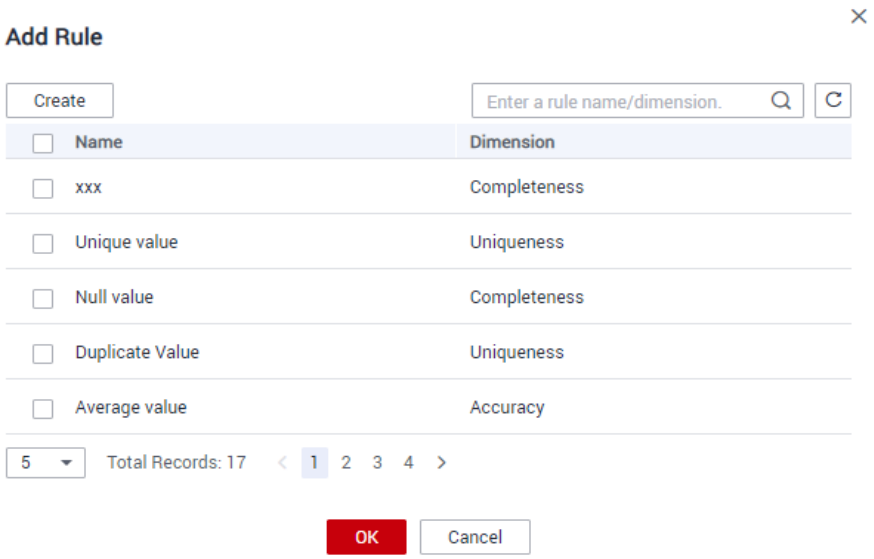
Step 5 In the dialog box displayed, click **Add Rule**.

Figure 5-189 Adding a quality rule



The **Add Rule** dialog box lists all default quality rules supported by DataArts Quality. Select a rule and click **OK**. If these quality rules cannot meet your requirements, you can customize one. In the **Add Rule** dialog box, click **Create** to navigate to DataArts Quality and create a rule on the page displayed. See **Creating Rule Templates**.

Figure 5-190 Add Rule dialog box

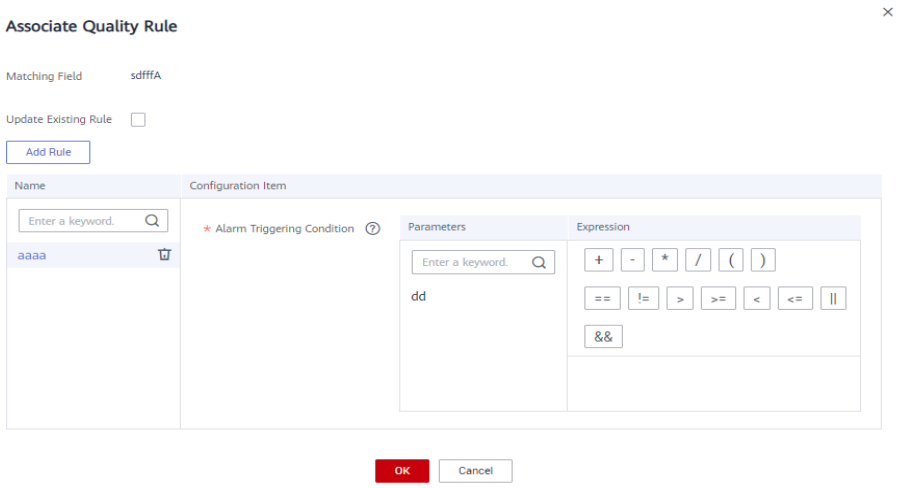


After a rule is added, the **Associate Quality Rule** dialog box is displayed. Select a rule from the rule name list, set **Alarm Condition**, and click **OK**.

- In the **Alarm Condition** text box, enter an expression. When a quality job is running, the system calculates the result of the alarm condition expression and determines whether to trigger the alarm based on the result of the expression. If the expression result is **true**, the alarm will be triggered. Otherwise, no quality alarm will be triggered.
- An alarm condition expression consists of alarm parameters and logical operators.

  The alarm parameters of each rule are displayed as buttons. If you click these buttons, the alarm conditions are expressed in the sequence of alarm parameters, such as *${1}*, *${2}*, and *${3}*. The variable names indicate the alarm parameters. In other words, when setting **Alarm Condition**, use the variable *${1}* to represent the first alarm parameter, *${2}* to represent the second alarm parameter, and so on.

Figure 5-191 Setting an alarm triggering condition

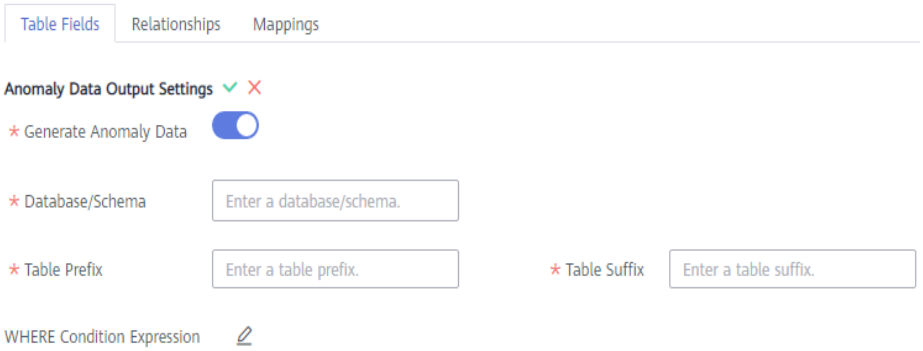


Step 6 (Optional) If you want to store anomaly data that does not comply with the preset rules in the exception table, enable **Anomaly Data Output Settings**.

Figure 5-192 Enabling Anomaly Data Output Settings

Click the pen icon next to **Anomaly Data Output Settings** and enable **Generate Anomaly Data**. The anomaly data will be stored in the specified database based on the settings.

Figure 5-193 Anomaly Data Output Settings



The parameters are as follows:

- **Database/Schema**: database or schema that stores anomaly data
- **Table Prefix**: prefix of the table that stores anomaly data
- **Table Suffix**: suffix of the table that stores anomaly data

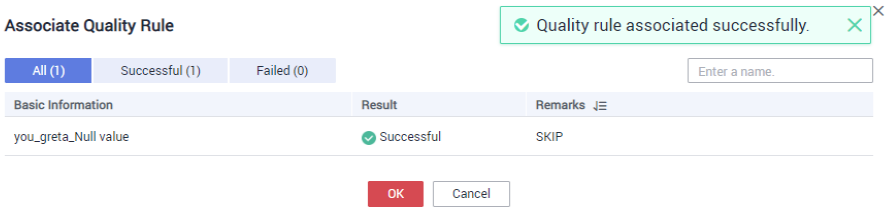
Click ✓ to save the settings.

Step 7 (Optional) By default, the quality rule applies to the entire table. If you want to query data in specified partitions, set the where condition.

Figure 5-194 Where condition

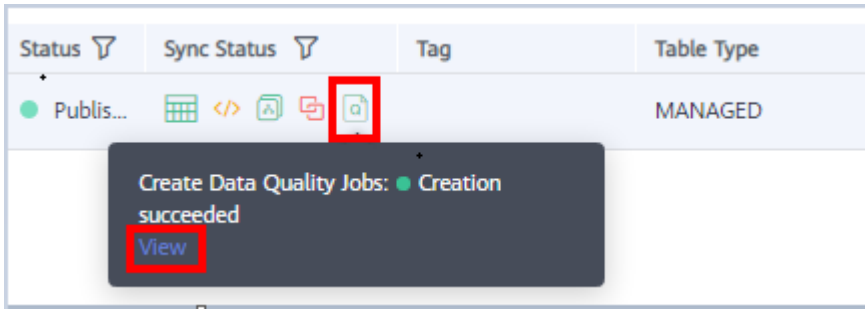
Step 8 View the association result. If the association is successful, click **OK**. If the association fails, find the failure cause, correct it, and associate the quality rule again.

Figure 5-195 Association results



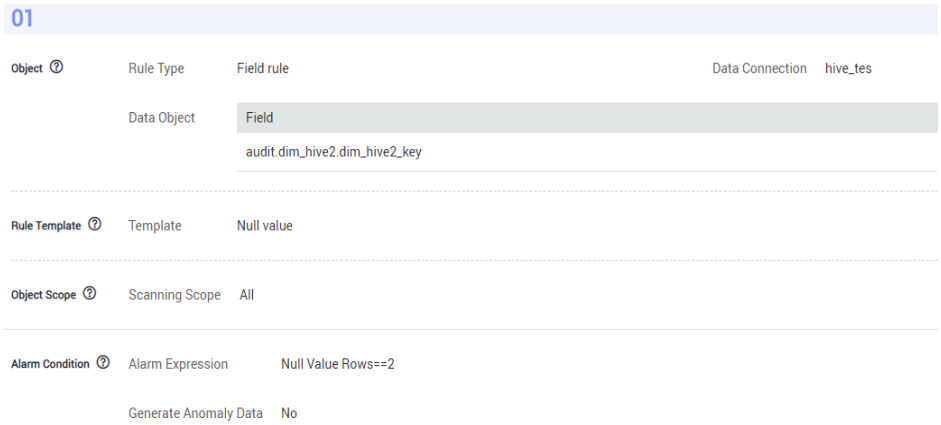
Step 9 Go back to the ER model list, locate the table that you just associated with a quality rule. In the **Sync Status** column, move your pointer to  and click **View**.

Figure 5-196 Quality job sync status



Step 10 On the page displayed, click the **Rule Configurations** tab to view the rule you just added.

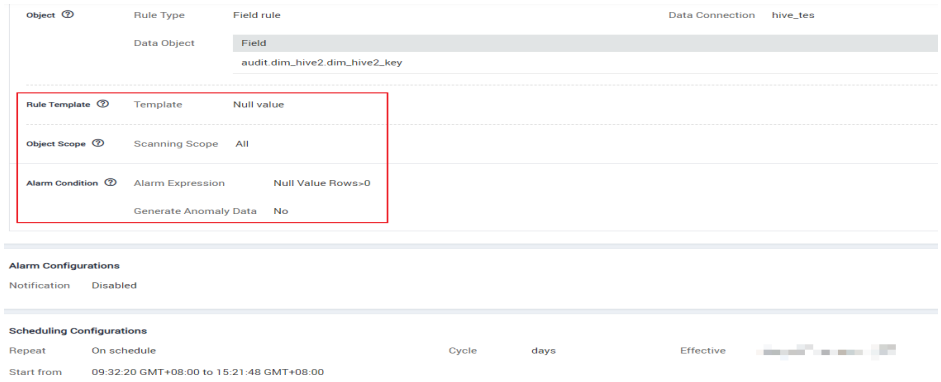
Figure 5-197 Quality rules



If a table is associated with a data standard when it is created, the corresponding quality rule is generated after the table is published. You can also view the rule on the **Quality Jobs** page.

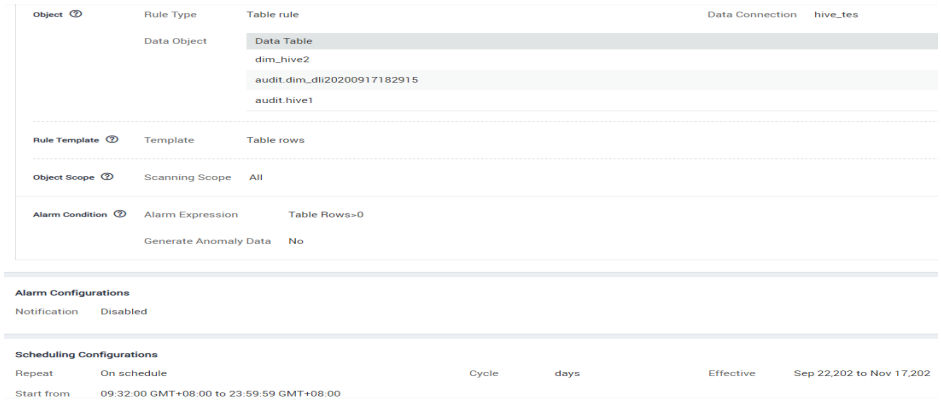
The following provides an example of the quality rule generated based on the data standard associated with a field:

Figure 5-198 Quality rule associated with a field



The following provides an example of the quality rule generated based on the data standard associated with a lookup table:

Figure 5-199 Quality rules for data standards



----End

5.9.5 Viewing Tables

Tables in an ER model can be displayed in the model view or list view. You can view table details, relationship diagrams, and publish history, as well as preview SQL statements.

Querying the Model View

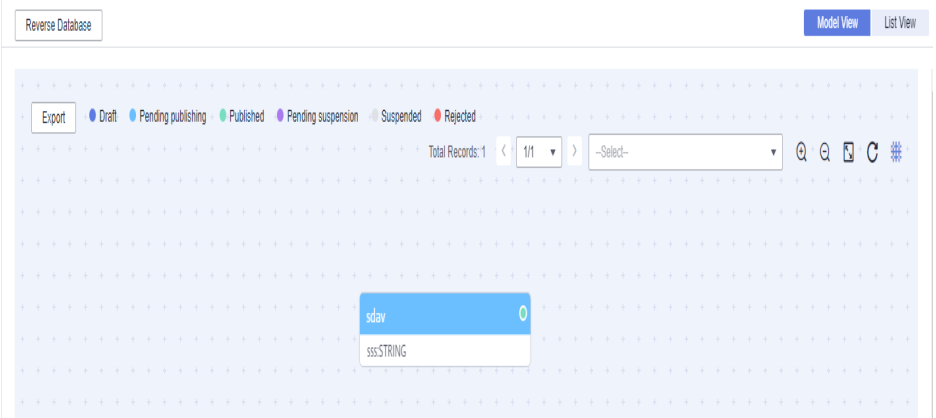
After creating a table in an ER model, you can query the table models in the list view or model view. The created tables are displayed in the list view by default. You can switch to the model view if you like.

Step 1 On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.

Step 2 In the ER model tree, select a model other than the OBS type, expand it, and select an object.

Step 3 On the **ER Modeling** page, all created tables are displayed in the list view by default. You can click the **Model View** in the upper right corner on the **ER Modeling** page to change the view mode. You can click **List View** to switch back to the table list.

Figure 5-200 Model view



The following functions are supported in the model view:

- Double-click a table name to view the table details.
- Click **Export** in the upper left corner to export the model view as an image.
- Enter a table name in the search box in the upper right corner to quickly find the table you want to view.
-  represents zoom in, zoom out, full screen, switch between physical and logical models, refresh, and canvas display, respectively.

----End

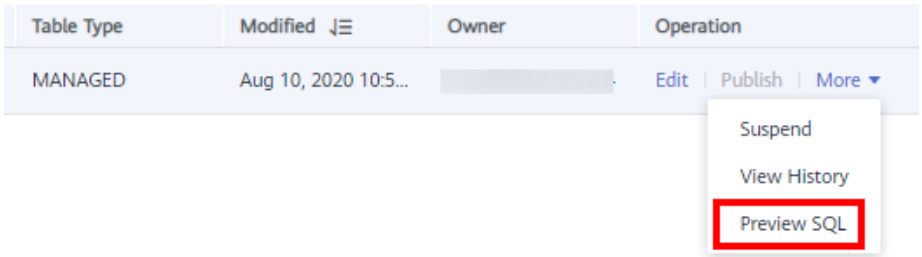
Viewing Table Details and Previewing an SQL Statement

Step 1 On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.

Step 2 On the model overview page, click the card of the required logical model, and select a subject in the subject directory. All tables under the subject are displayed in the list on the right.

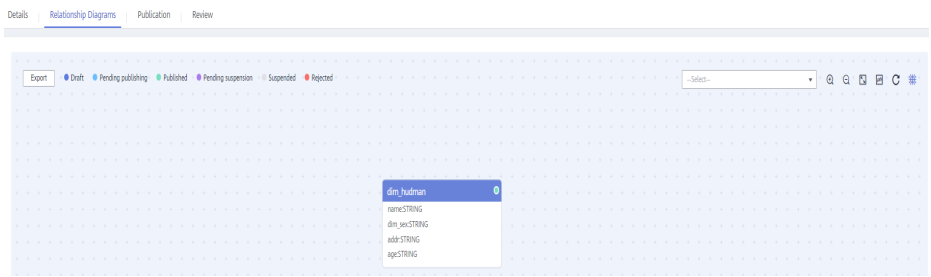
Step 3 In the table list, select a table, and choose **More** > **Preview SQL** in the **Operation** column to preview or copy the SQL statement. Then, click **OK** to return to the previous page.

Figure 5-201 ER model list



Step 4 In the table list, click a table name to access the table details page and view the table details, relationship diagrams, publish history, and review history.

Figure 5-202 Relationship diagrams



----End

Viewing Publish History

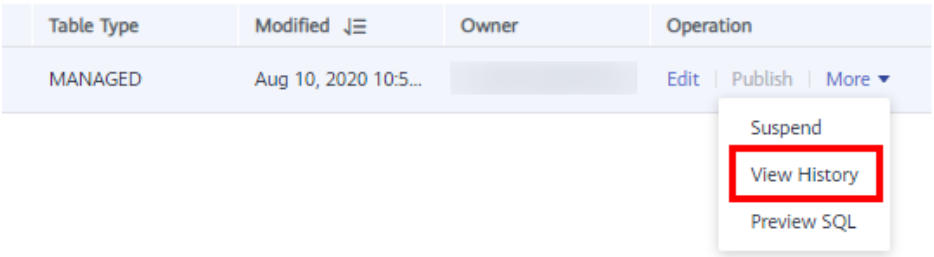
After a table is published, you can view its publish history, version comparisons, and publish logs. If a table fails to be published, or a data asset or data quality job fails to be synchronized, you can view the publish log to troubleshoot the fault and publish or synchronize it again.

Step 1 On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.

Step 2 On the model overview page, click the card of the required logical model, and select a subject in the subject directory. All tables under the subject are displayed in the list on the right.

Step 3 In the table list, locate the target table, and choose **More** > **View History**. On the page displayed, you can view the table publish history, version comparisons, and publish logs.

Figure 5-203 Viewing publish history



----End

5.9.6 Modifying Subjects, Directories, and Processes

Modifying Subjects in Batches

Currently, only subjects of information architectures, ER models, dimensions, fact tables, summary tables, and technical metrics can be modified in batches. The modification procedure is similar.

This section describes how to modify the subject of information architecture in batches.

Step 1 On the DataArts Architecture page, choose **Information Architecture** in the left navigation pane.

Step 2 On the page displayed, select the targets whose subjects need to be modified, and choose **More** > **Modify Subject**. After the configuration is complete, click **OK**.

Figure 5-204 Modifying subjects

----End

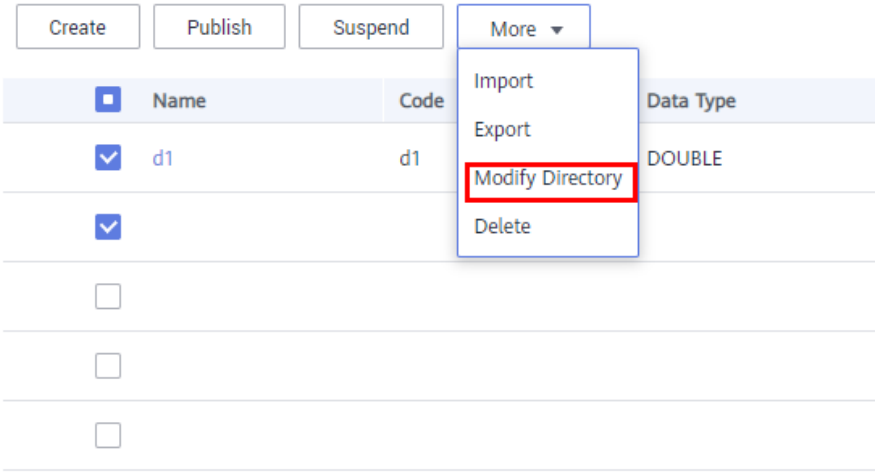
Modifying Directories in Batches

Currently, only directories of lookup tables and data standards can be modified in batches.

Step 1 On the DataArts Architecture page, choose **Standards** > **Lookup Tables** or **Standards** > **Data Standards** in the left navigation pane.

Step 2 On the page displayed, select the targets whose directories need to be modified, and choose **More** > **Modify Directory**.

Figure 5-205 Modifying directories of lookup tables in batches



----End

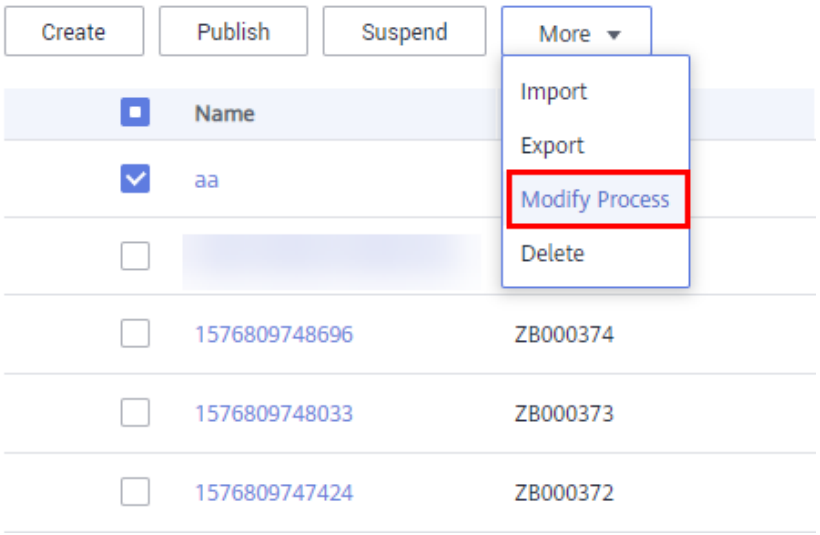
Modifying Processes in Batches

Currently, only the processes of business metrics can be modified in batches.

Step 1 On the DataArts Architecture page, choose **Metrics** > **Business Metrics** in the left navigation pane.

Step 2 On the page displayed, select the metrics whose processes need to be modified, and choose **More** > **Modify Process**.

Figure 5-206 Modifying processes



----End

5.9.7 Review Center

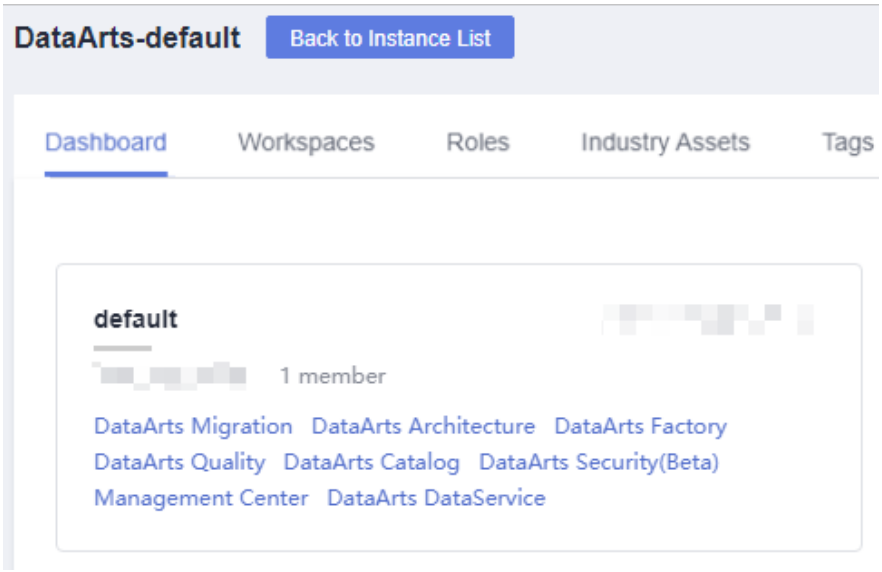
After the modeling and data processing tasks generated in the development environment are submitted, they are stored in the review center. After the tasks are approved on the **Review Center** page, these tasks are available in the production environment.

Reviewer's Audit Objects

If you are a reviewer, use the reviewer account with caution.

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

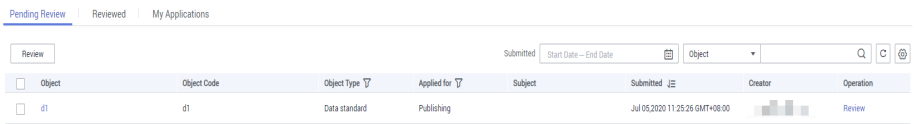
Figure 5-207 DataArts Architecture



2. Choose **Metrics** > **Review Center** in the left navigation bar, click the **Pending Review** tab, find the object to be reviewed in the list, and click **Review** on the right.

   You can also select multiple objects to be reviewed and click **Review** in the upper left corner to review them in batches.

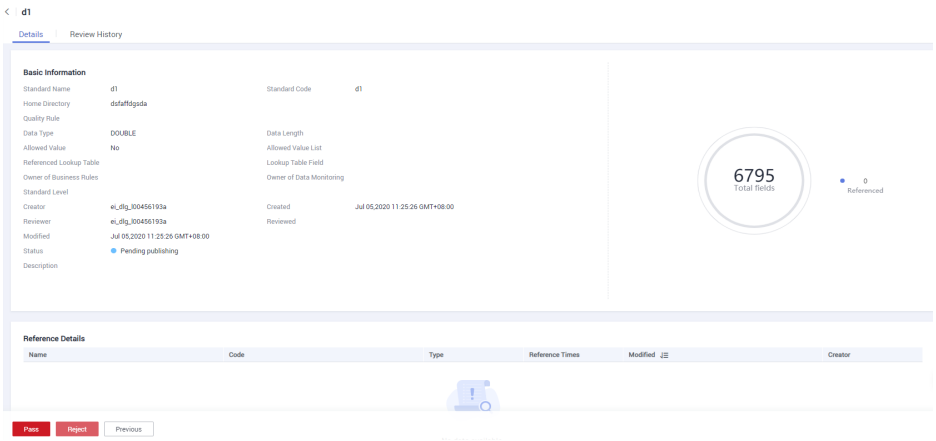
Figure 5-208 Pending Review tab page



3. On the page displayed, confirm the information and click **Accept**. In the dialog box displayed, enter the review comments and click **OK**.

   If the information is incorrect, click **Reject**. In the dialog box displayed, enter the reasons for rejecting the application and click **OK**.

Figure 5-209 Review Information area



Pending Review, Reviewed, and My Applications Tab Pages

- **Pending Review** tab page

  On the DataArts Architecture console, choose **Metrics** > **Review Center** in the left navigation pane and click the **Pending Review** tab. On the page displayed, you can view the applications to be reviewed.

- **Reviewed** tab

  On the DataArts Architecture console, choose **Metrics** > **Review Center** in the left navigation pane and click the **Reviewed** tab. On the page displayed, you can view the applications that have been approved.

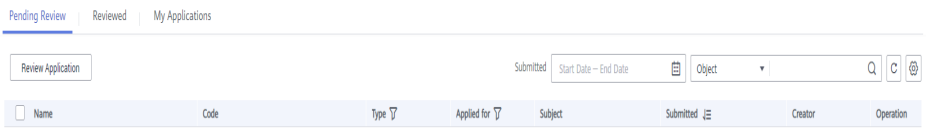
- **My Applications** tab

  On the DataArts Architecture console, choose **Metrics** > **Review Center** in the left navigation pane and click the **My Applications** tab. On the page displayed, you can view the applications that you have submitted.

Pending Review

Step 1 On the DataArts Architecture console, choose **Metrics** > **Review Center** in the left navigation pane. The **Pending Review** tab page is displayed by default.

Figure 5-210 Pending Review tab page



Function Area	Description
1	Batch Review 1. Select multiple pieces of information to be reviewed. 2. Click **Review Application**. 3. In the dialog box displayed, enter the valid review comments. 4. Click **Accept** to approve the selected targets in batches, or click **Reject** to reject the selected targets in batches.
2	Single Review 1. Click **Review** in the **Operation** column. The page for reviewing the information is displayed. 2. Select the review result and enter valid review comments based on service requirements. 3. Click **OK**.

Function Area	Description
3	•  allows you to specify a time range during which the information to be viewed is displayed. •  allows you to query the to-be-reviewed information about objects and creators. •  allows you to set the headers of tables to be reviewed. •  allows you to refresh the current page.

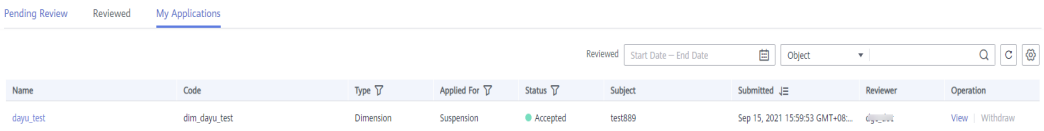
----End

My Applications

Step 1 On the DataArts Architecture console, choose **Metrics** > **Review Center** in the left navigation pane.

Step 2 Click **My Applications**.

Figure 5-211 My Applications tab page



You can perform the following operations:

- Click **View** in the **Operation** column to view information about a specified row.
- Click **Withdraw** in the **Operation** column to withdraw the application.

----End

5.10 Tutorials

5.10.1 DataArts Architecture Example

DataArts Architecture can be used to create entity-relationship (ER) models and dimensional models to standardize and visualize data development and output data governance methods that can guide development personnel to work with ease.

This section covers the following scenarios:

- Design a data model for the taxi travel data in an MRS Hive data lake.
- The original taxi travel data table **sdi_taxi_trip_data** is stored in the **demo_sdi_db** database.
- The following table lists the data fields in the original data table **sdi_taxi_trip_data**.

  The following table lists the taxi trip data:

Table 5-66 Taxi trip data

No.	Field Name	Field Description
1	VendorID	Vendor ID. Possible values are: 1=A Company 2=B Company
2	tpep_pickup_datetime	Time when a passenger gets on a taxi.
3	tpep_dropoff_datetime	Time when a passenger gets off a taxi.
4	passenger_count	Number of passengers.
5	trip_distance	Driving distance.
6	ratecodeid	Charge rate code. Possible values are: 1=Standard rate 2=JFK 3=Newark 4=Nassau or Westchester 5=Negotiated fare 6=Group ride
7	store_fwd_flag	Store-and-forward flag.
8	PULocationID	Location at which a passenger gets on a taxi.
9	DOLocationID	Location at which a passenger gets off a taxi.

No.	Field Name	Field Description
10	payment_type	Payment type. Possible values are: 1=Credit card 2=Cash 3=No charge 4=Dispute 5=Unknown 6=Voided trip
11	fare_amount	Fare amount.
12	extra	Extra fee.
13	mta_tax	MTA tax.
14	tip_amount	Tip amount.
15	tolls_amount	Toll amount.
16	improvement_surcharge	Improvement surcharge.
17	total_amount	Total amount.

The process of using DataArts Architecture is as follows:

1. **Preparations**
   - **Add reviewers**: In the DataArts Architecture module, all business processes must be approved. Therefore, add reviewers first before conducting any operations. Only the workspace admin has the permissions required to add reviewers.
   - **Configuration Center** provides abundant custom options. You can customize the configuration to meet your demands.
2. **Data Survey**: A data survey involves collecting data that is generated when sorting business requirements, creating business processes, and classifying data subjects based on the existing business data and industry status.
   - **Subject design** is a hierarchical architecture that classifies and defines data to help clarify data assets and specify relationships between business domains and business objects.
   - **Process design**: This example does not contain this. Process design is to generate a structured framework of data processing process, including the categories, levels, boundaries, scope, and input/output relationships, and reflect the business models and characteristics of your enterprise.
3. **Standards**: Create lookup tables and data standards.
   - **Create and publish a lookup table**: A lookup table includes a series of allowed values and additional text descriptions that are generally associated with data standards to generate a range of values for the verification of quality monitoring rules.

- **Create and publish a data standard**: A data standard refers to the description of attribute data meanings and business rules that enterprises must comply with. It describes the common understanding of certain data at the company level.

4. **Models**: Use ER modeling and dimensional modeling methods to perform hierarchical modeling.
   - **ER modeling: Create a model at the SDI and DWI layers, respectively.**
     - **SDI** stands for Source Data Integration and is the source data layer. SDI is a simple implementation of source system data.
     - **DWI** stands for Data Warehouse Integration, also called the data consolidation layer. DWI integrates and cleans data from multiple source systems, and implements entity relationship modeling based on the three normal forms.
   - **Dimensional modeling: Create and publish a dimension at the DWR layer. & Creating and Publishing a Fact Table for the DWR Layer.**
     - **Data Warehouse Report (DWR)** is based on the multi-dimensional model and its data granularity is the same as that of the DWI layer.
     - **Dimension** is the perspective to observe and analyze business data and assist in data aggregation, drilling, slicing, and analysis, and used as a GROUP BY condition in SQL statements.
     - A **fact table** that belongs to a business process can enrich the affair information corresponding to the specific business process.
5. **Metric design: Create and publish a technical metric**: Create and publish a business metric (not involved in this example) and a technical metric. Technical metrics are classified into atomic, derivative, and compound metrics.
   - A **metric** consists of its name and value. The metric name and its definition reflect the quality and quantity of the metric. The metric value reflects the quantifiable values of the specified time, location, and condition of the metric.

     Business metrics are used to guide technical metrics, and technical metrics are used to implement business metrics.
   - **Atomic metrics** are generated based on dimension tables and fact tables of a multidimensional model. The objects and the finest data granularity of an atomic metric are consistent with those of the multidimensional model.

     An atomic metric usually consists of measures and attributes related with measures and business objects, all of which aim to support agile self-service consumption of the metric.
   - **Derivative metrics** are aggregated from the definitions, modifiers, and dimensions of atomic metrics. Therefore, their definitions, modifiers, and dimensions are derived from the attributes of atomic metric associated tables as well.
   - **Compound metrics** are generated by adding one or more derivative metrics. The dimensions and modifiers of a compound metric are the same as those of the derivative metrics.

     New dimensions and modifiers cannot be generated outside the scope of derivative metrics, dimensions, and modifiers.

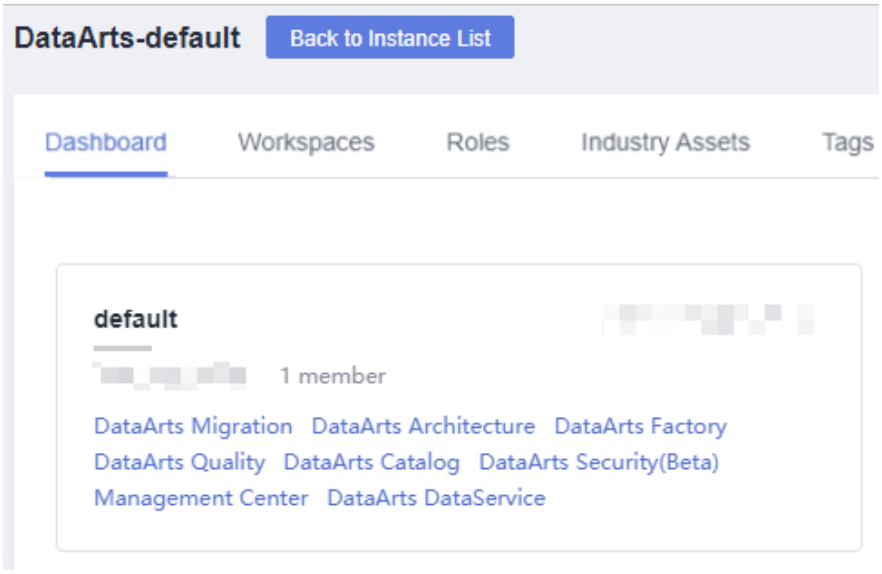
6. **Dimensional modeling: Create and publish a summary table at the DM layer**.
   - **Data Mart (DM)** is where multiple types of data are summarized. DM is designed to display the summarized data.
   - A **summary table** consists of specific analysis objects (for example, members) and related statistical metrics. The statistical metrics included in a summary table have the same statistical granularity (for example, members). The summary table provides users with all statistics-granularity-themed data (such as a member theme market).

Adding Reviewers

In the DataArts Architecture module, all modeling steps must be reviewed. Therefore, you need to add a reviewer first. DAYU Administrator or the workspace administrator has the permission to add reviewers.

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Architecture**.

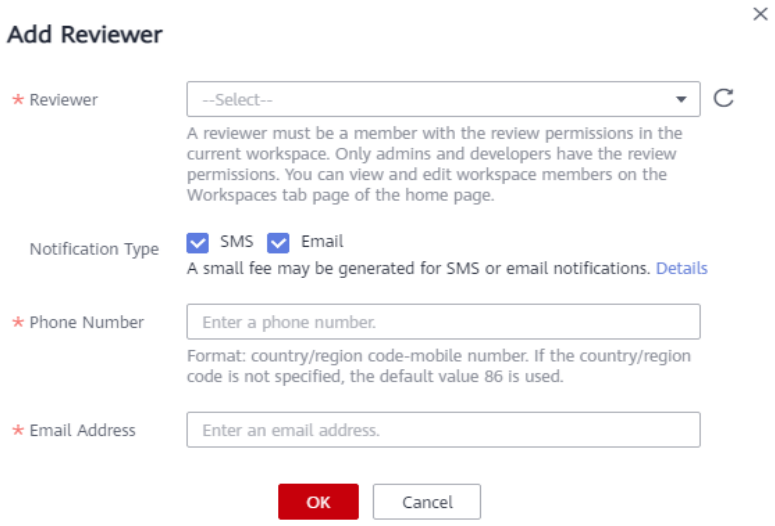
   Figure 5-212 DataArts Architecture



2. In the navigation pane on the left, choose **Configuration Center**. On the displayed **Reviewers** page, click **Add**.
3. Select a reviewer (administrator or developer), enter the correct email address and phone number, and click **OK**.

   You can also add your current account as a reviewer. In this way, auto review is supported in subsequent operations. Add more reviewers, if required.

Figure 5-213 Adding a reviewer

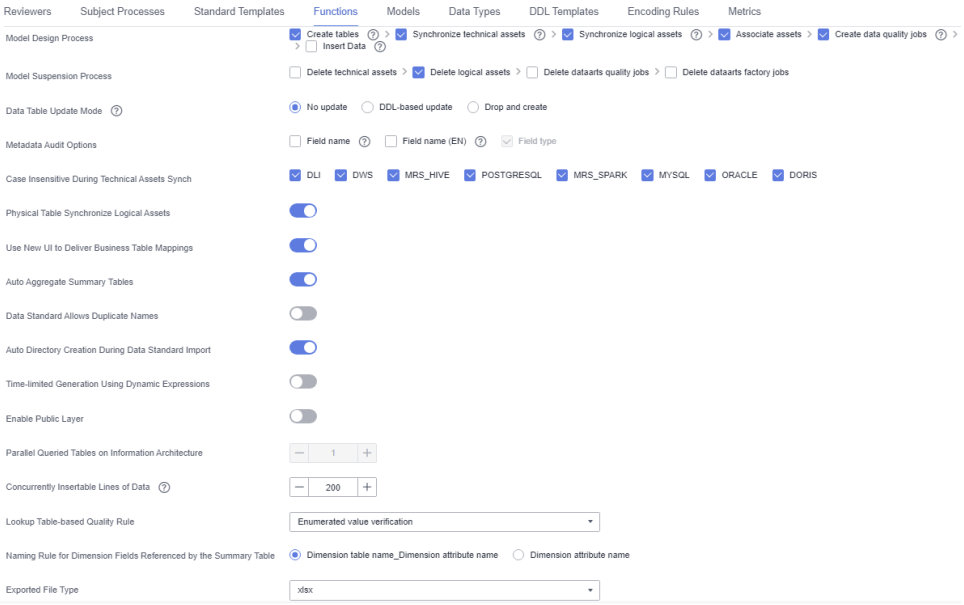


Configuration Center Management

DataArts Architecture configuration center provides abundant custom options. You can customize the configuration to meet your demands.

1. On the DataArts Architecture console, choose **Configuration Center** in the navigation pane on the left.
2. Click the **Functions** tab and set **Model Design Process**.

Figure 5-214 Functions



3. Click **OK**.

Designing a Subject

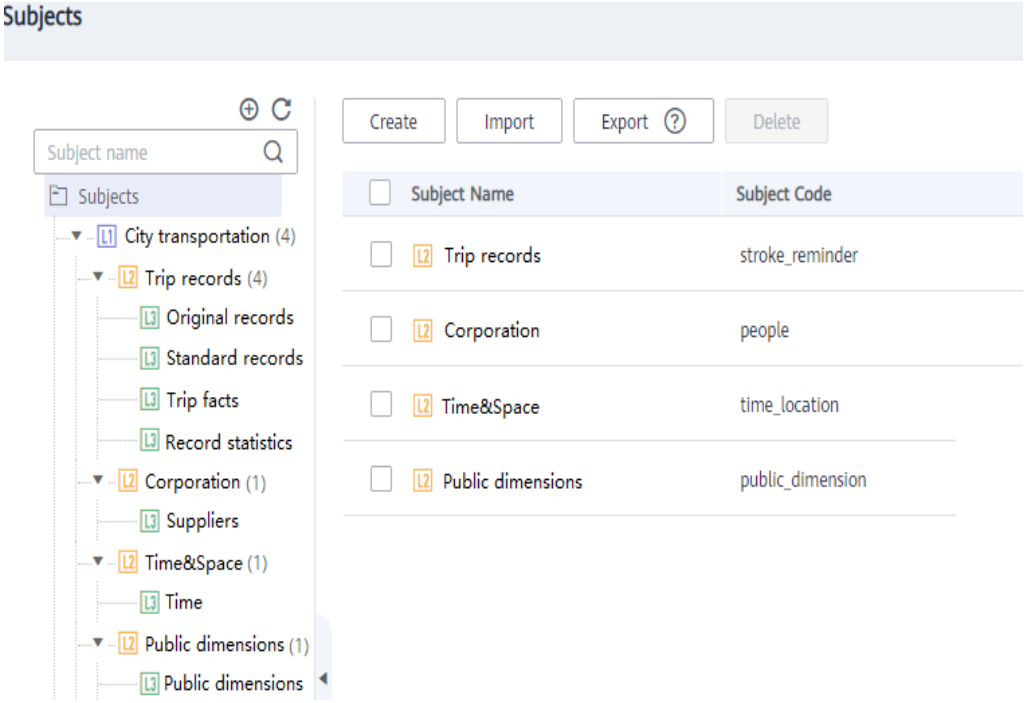
This section uses the subjects listed in **Table 5-67** as an example.

- There is a subject area group named **City transportation**.
- Under **City transportation**, there are four subject areas: **Trip records**, **Corporation**, **Time&Space**, and **Public dimensions**.
- Under **Trip records**, there are four business objects: **Original records**, **Standard records**, **Trip facts**, and **Record statistics**.
- Under **Corporation**, there is one business object: **Suppliers**.
- Under **Time&Space**, there is one business object: **Time**.
- Under **Public dimensions**, there is one business object: **Public dimensions**.

Table 5-67 Subject design

Subject Area Group Name (L1)	Subject Area Group Code (L1)	Subject Area Name (L2)	Subject Area Code (L2)	Business Object Name (L3)	Business Object Code (L3)
City transportation	city_traffic	Trip records	stroke_reminder	Original records	origin_stroke
				Standard records	stand_stroke
				Trip facts	stroke_fact
				Record statistics	stroke_statistic
		Corporation	people	Suppliers	vendor
		Time&Space	time_location	Time	date
		Public dimensions	public_dimension	Public dimensions	public_dimension

Figure 5-215 Designing a subject



Procedure

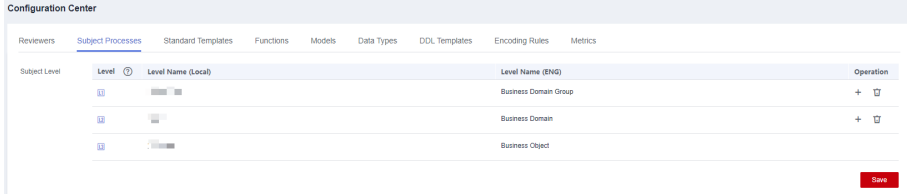
Step 1 Log in to the DataArts Studio console. Locate the created DataArts Studio instance and click **Access**.

Step 2 In the workspace list, locate the target workspace and click **DataArts Architecture**.

Step 3 Choose **Configuration Center** in the navigation pane on the left. Click the **Subject Processes** tab, and use the default three levels.

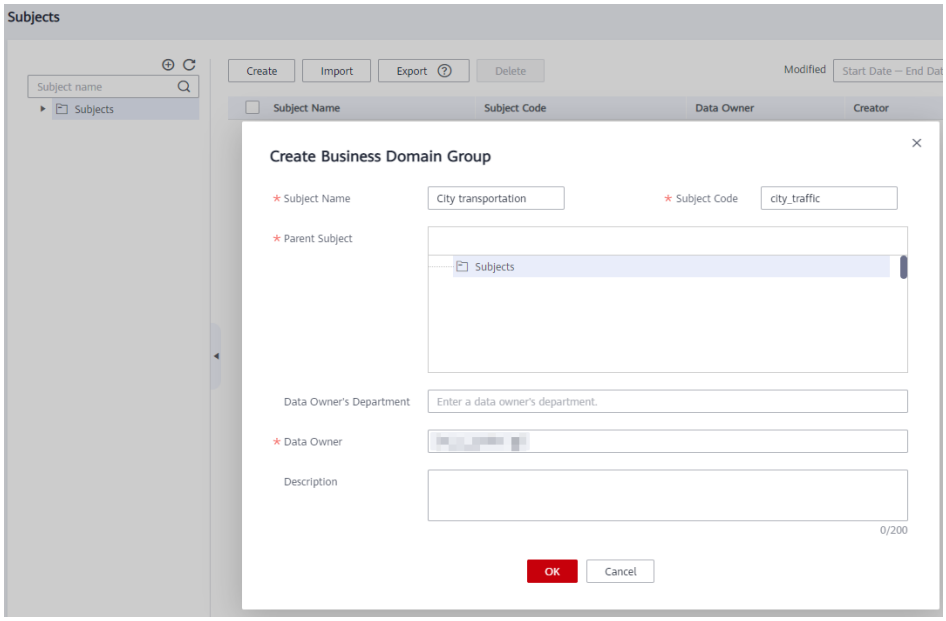
There can be a maximum of seven subject levels, a minimum of two subject levels, and three subject levels by default. L1 to L7 are used to represent the layers. The last level is **Business Object** and cannot be customized. The names of other levels can be customized. The levels configured in **Configuration Center** take effect on the **Subjects** page.

Figure 5-216 Configuring the subject levels



Step 4 On the DataArts Architecture console, choose **Data Survey** > **Subjects** in the left navigation pane. On the page displayed, click **Create** to create an L1 subject, which is a subject area group.

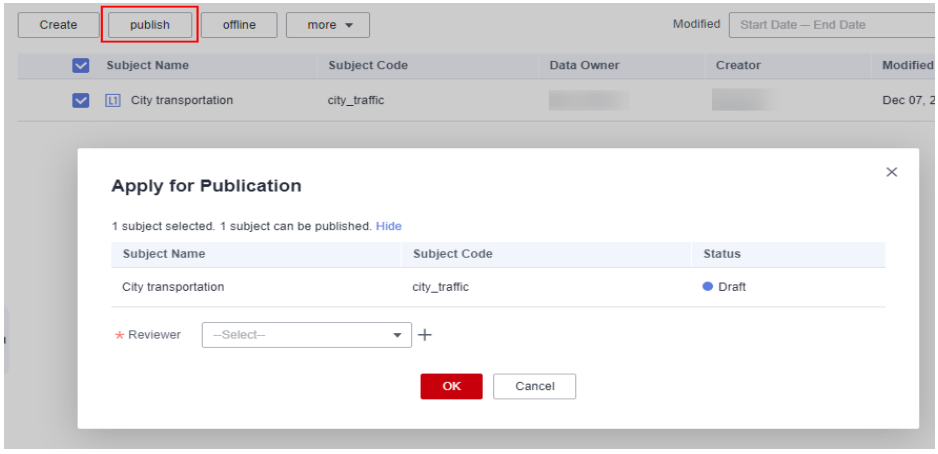
Figure 5-217 Creating an L1 subject



In the dialog box displayed, set the parameters as shown in **Figure 5-217** and click **OK**.

Step 5 Select the created subject area group and click **publish**. In the **Apply for Publication** dialog box, select a reviewer and click **OK**. Wait for the reviewer to review the application. If you have the reviewer permissions, select **Auto-review** and click **OK**.

Figure 5-218 Publishing a subject area group

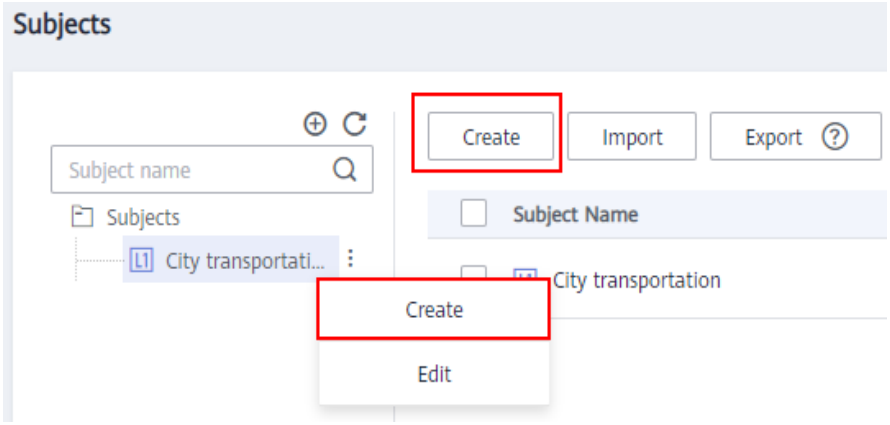


Step 6 Create four L2 subjects under the L1 subject **City transportation**: **Trip records**, **Corporation**, **Time&Space**, and **Public dimensions**.

Perform the following procedure to create a subject area named **Trip records**. The procedure for creating other subject areas is similar.

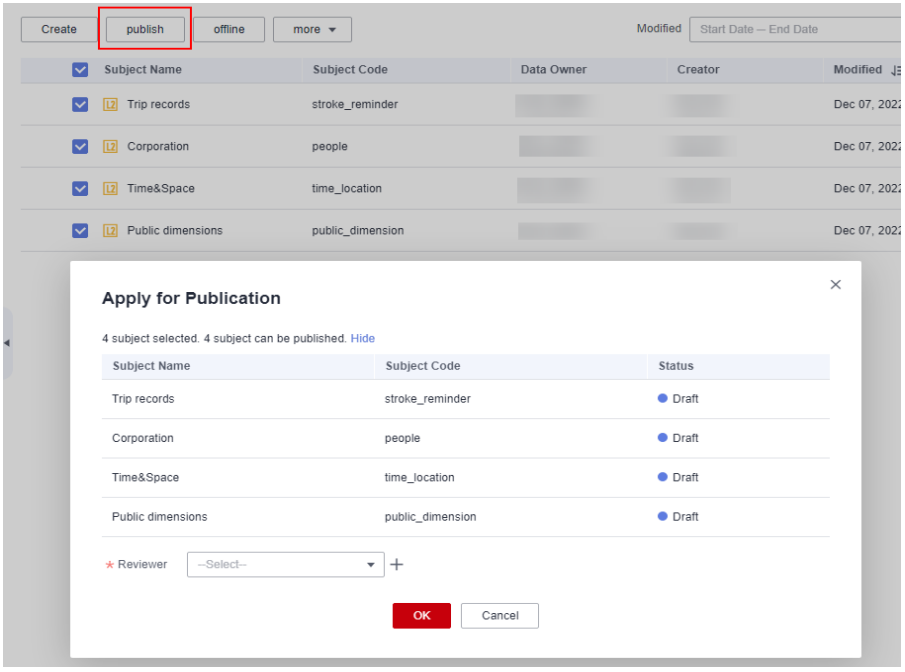
1. Right-click the L1 subject **City transportation** in the subject tree, and select **Create** from the shortcut menu. Alternatively, click **Create** in the right pane.

Figure 5-219 Creating an L2 subject



2. In the dialog box displayed, set **Subject Name** and **Subject Code** to the values of **Subject Area Name** and **Subject Area Code** in **Table 5-67**, set other parameters based on project requirements, and click **OK**.
3. Select the created subject area and click **publish**. In the **Apply for Publication** dialog box, select a reviewer and click **OK**. Wait for the reviewer to review the application. If you have the reviewer permissions, select **Auto-review** and click **OK**.

Figure 5-220 Publishing a subject area



Step 7 Create business objects.

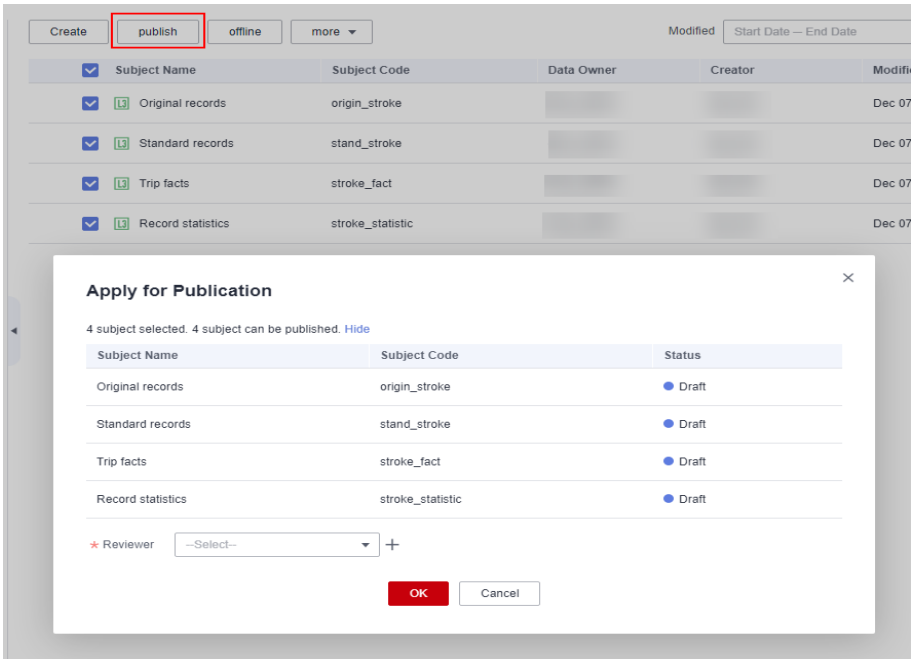
- Under **Trip records**, create four business objects: **Original records**, **Standard records**, **Trip facts**, and **Record statistics**.
- Under **Corporation**, create one business object: **Suppliers**.

- Under **Time&Space**, create one business object: **Time**.
- Under **Public dimensions**, create one business object: **Public dimensions**.

Perform the following procedure to create a business object named **Original records** in the subject area **Trip records**. The procedure for creating other business objects is similar.

1. Right-click the L2 subject **Trip records** in the subject tree, and select **Create** from the shortcut menu. Alternatively, click **Create** in the right pane.
2. In the dialog box displayed, set **Subject Name** and **Subject Code** to the values of **Business Object Name** and **Business Object Code** in **Table 5-67**, set other parameters based on project requirements, and click **OK**.
3. Select the created business object and click **publish**. In the **Apply for Publication** dialog box, select a reviewer and click **OK**. Wait for the reviewer to review the application. If you have the reviewer permissions, select **Auto-review** and click **OK**.

Figure 5-221 Publishing a business object



----End

Creating and Publishing Lookup Tables

This section uses the lookup tables listed in **Table 5-68** as an example.

Table 5-68 Lookup tables

Directory	*Table Name	* Table English Name	Table Description	* Field Name	* Field Code	* Data Type	Field Description
payment_type	payment_type	payment_type	None	payment_type_id	payment_type_id	BIGINT	None
				payment_type_value	payment_type_value	STRING	None
vendor	vendor	vendor	None	vendor_id	vendor_id	BIGINT	None
				vendor_value	vendor_value	STRING	None
rate	rate_code	rate_code	None	rate_code_id	rate_code_id	BIGINT	None
				rate_code_value	rate_code_value	STRING	None

Procedure

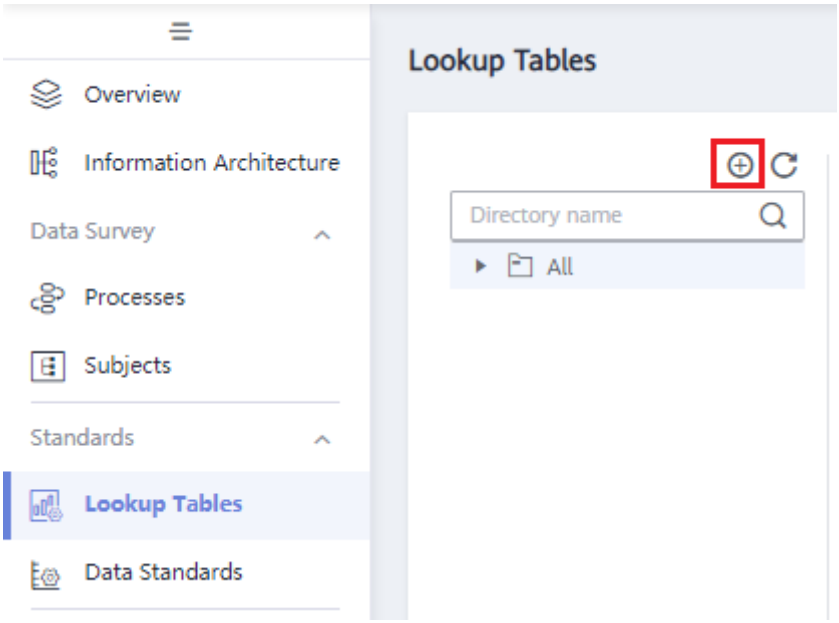
Step 1 On the DataArts Architecture console, choose **Standards** > **Lookup Tables** in the navigation pane on the left.

Step 2 Create three lookup table directories: **payment_type**, **vendor**, and **rate**.

Perform the following procedure to create a directory named **payment_type**. The procedure for creating other directories is similar.

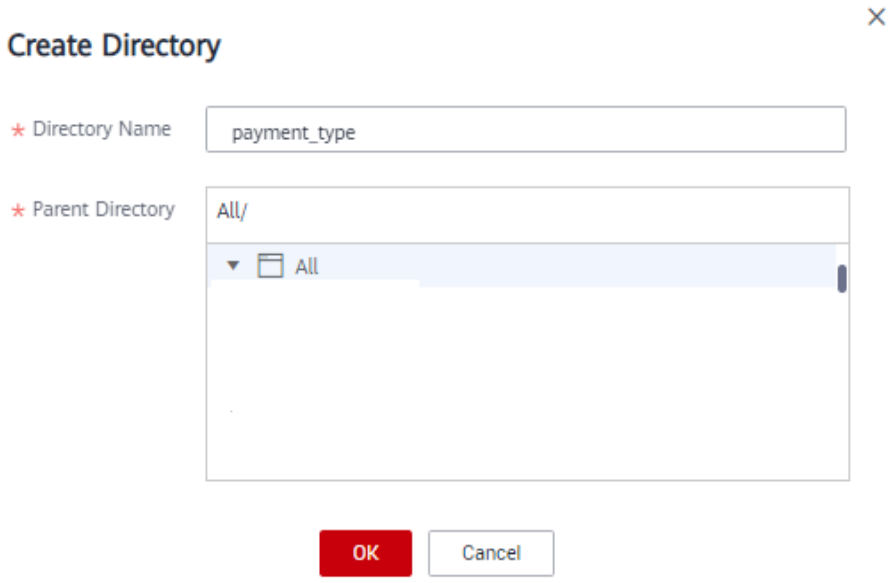
1. On the **Lookup Tables** page, click above the directory tree to create a directory.

Figure 5-222 Lookup table directory tree



2. In the dialog box displayed, enter a directory name, select a parent directory, and click **OK**.

Figure 5-223 Creating a directory for lookup tables

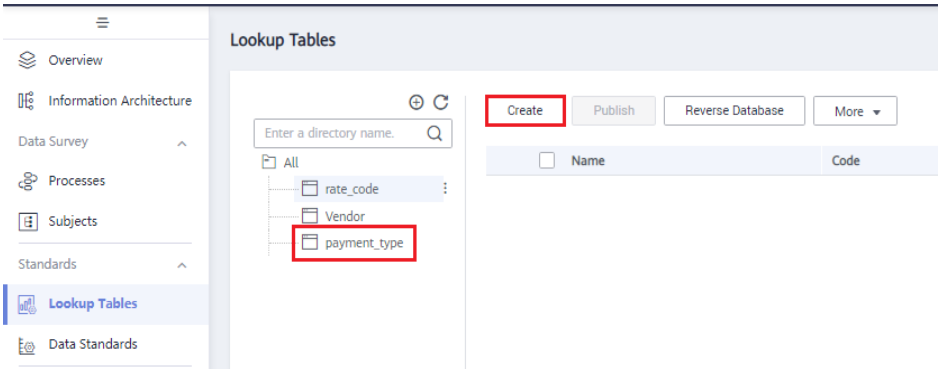


Step 3 Create three lookup tables: **payment_type**, **vendor**, and **rate_code**.

Perform the following procedure to create a lookup table named **payment_type**. The procedure for creating other lookup tables is similar.

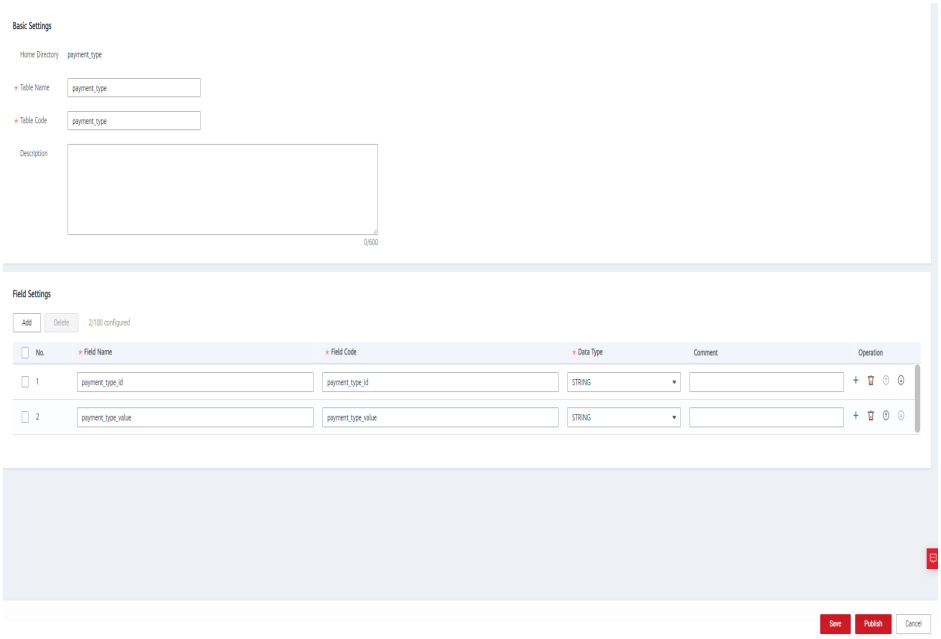
1. On the **Lookup Tables** page, click **payment_type** in the directory tree, and click **Create** on the page displayed.

Figure 5-224 Lookup Tables page



2. Set the parameters based on **Table 5-68** and click **Save**.

Figure 5-225 Creating a lookup table



3. Refer to **Step 3.1** to **Step 3.2** to create the lookup table **vendor** in the **vendor** directory and the lookup table **rate_code** in the **rate** directory.

Figure 5-226 Creating a lookup table named vendor

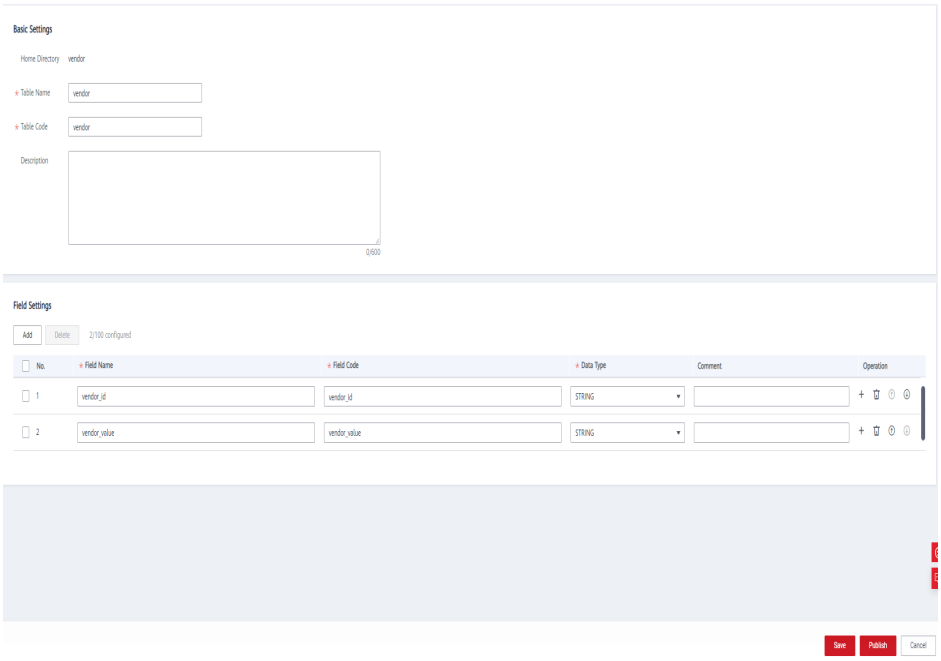
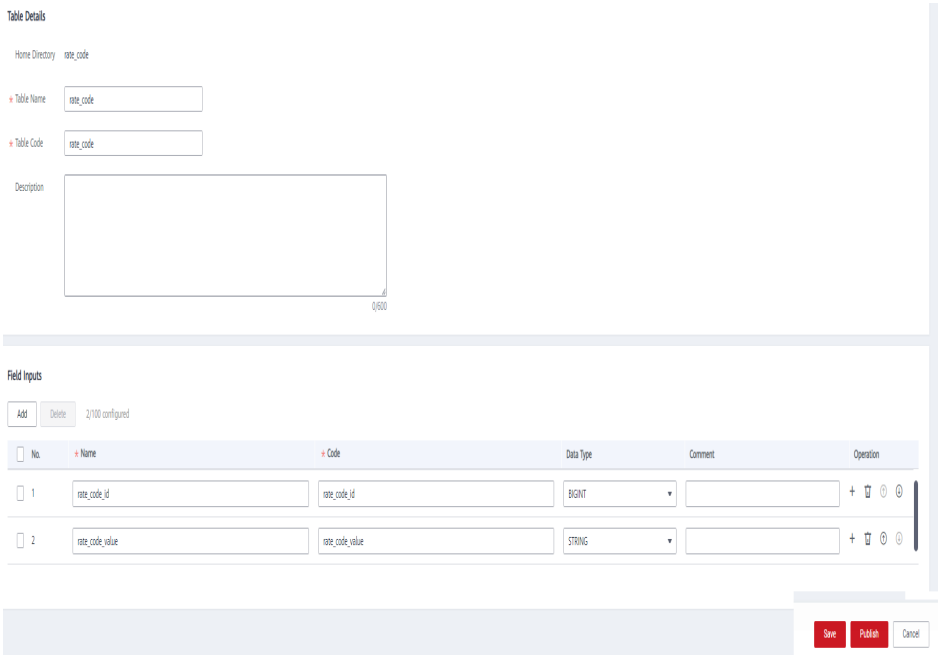


Figure 5-227 Creating a lookup table named rate_code



Step 4 Enter values for the three lookup tables **payment_type**, **vendor**, and **rate_code**.

On the **Lookup Tables** page, locate the row that contains the lookup table **payment_type**, and choose **More** > **Manage Value** in the **Operation** column. On the page displayed, click **Add** to add the values listed in **Table 5-69**.

Table 5-69 Values to be added for the lookup table payment_type

payment_type_id	payment_type_value
1	Credit card
2	Cash
3	No charge
4	Dispute
5	Unknown
6	Voided trip

Return to the **Lookup Tables** page, locate the row that contains the lookup table **vendor**, and choose **More** > **Manage Value** in the **Operation** column. On the page displayed, click **Add** to add the values listed in **Table 5-70**.

Table 5-70 Values to be added for the lookup table vendor

vendor_id	vendor_value
1	A Company
2	B Company

Return to the **Lookup Tables** page, locate the row that contains the lookup table **rate_code**, and choose **More** > **Manage Value** in the **Operation** column. On the page displayed, click **Add** to add the values listed in **Table 5-71**.

Table 5-71 Values to be added for the lookup table rate_code

rate_code_id	rate_code_value
1	Standard rate
2	JFK
3	Newark
4	Nassau or Westchester
5	Negotiated fare
6	Group ride

Step 5 Return to the **Lookup Tables** page, select the three lookup tables, and click **Publish**.

Step 6 In the **Apply for Publication** dialog box, select a reviewer and click **OK**. Wait for the reviewer to review the application. If you have the reviewer permissions, select **Auto-review** and click **OK**.

----End

Creating and Publishing Data Standards

In this example, you need to create the three data standards listed in **Table 5-72**.

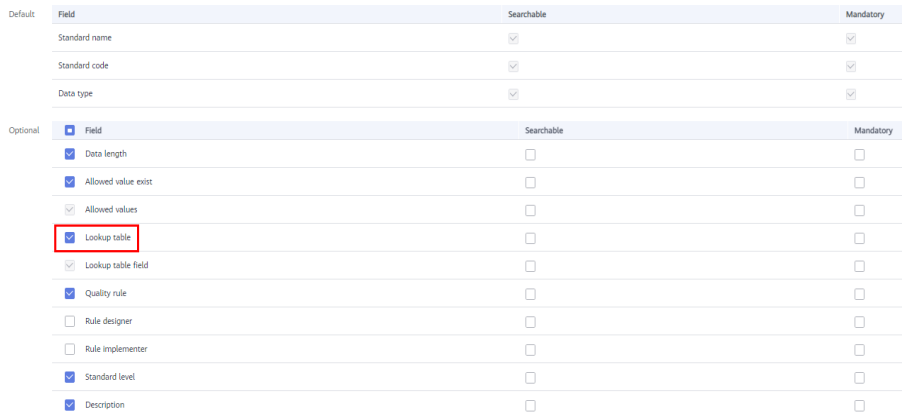
Table 5-72 Data standards

Directory	*Standard Name	* Standard Code (Custom)	*Data Type	Data Length	Lookup Table	*Lookup Table Field	Description
payment_type	payment_type	payment_type	Long integer (BIGINT)	None	payment_type	payment_type_id	None
vendor	vendor	vendor	Long integer (BIGINT)	None	vendor	vendor_id	None
rate	rate_code	rate_code	Long integer (BIGINT)	None	rate_code	rate_code_id	None

Step 1 On the DataArts Architecture console, choose **Standards** > **Data Standards** in the navigation pane on the left.

Step 2 If you access the Data Standards page for the first time, you must customize a template. The custom template can be modified in Configuration Center. Additionally, select **Lookup table**, as shown in the following figure.

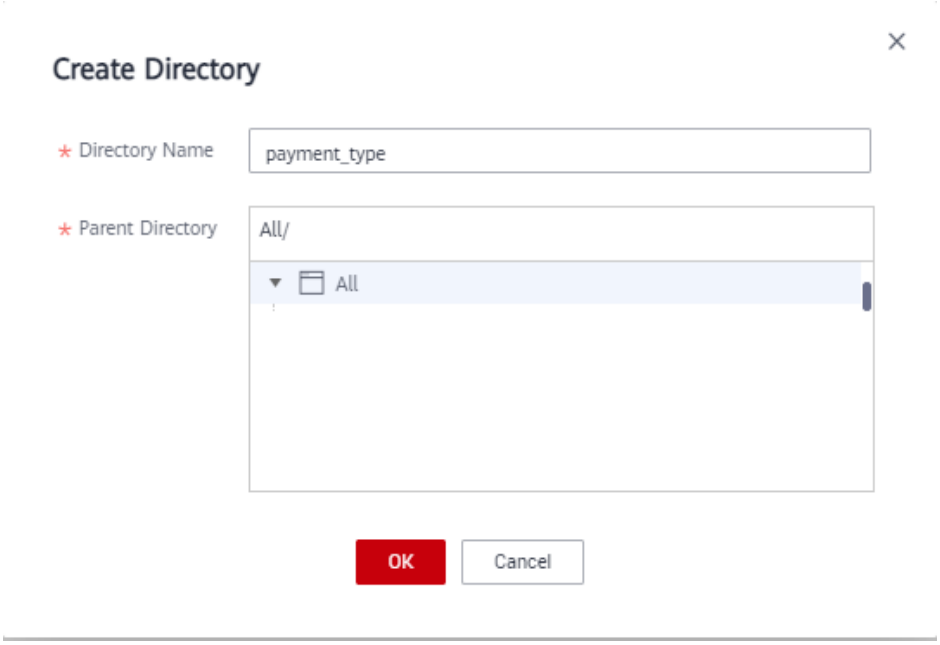
Figure 5-228 Customize Template



Step 3 Create three directories for data standards: **payment_type**, **vendor**, and **rate_code**.

In the upper part of the directory tree on the **Data Standards** page, click . In the dialog box displayed, enter the directory name as **payment_type**, select a parent directory, and click **OK**.

Figure 5-229 Creating a directory for data standards



Step 4 Create three data standards: **payment_type**, **vendor**, and **rate_code**.

1. In the directory tree on the **Data Standards** page, select the required directory and click **Create** on the page displayed on the right.
2. On the **Create Data Standard** page, configure the three data standards by referring to the following figures, and click **Save**. In this example, only a few parameters are selected for the data standard template. You can customize a data standard template by referring to **Configuration Center**.

Figure 5-230 Creating a data standard named payment_type

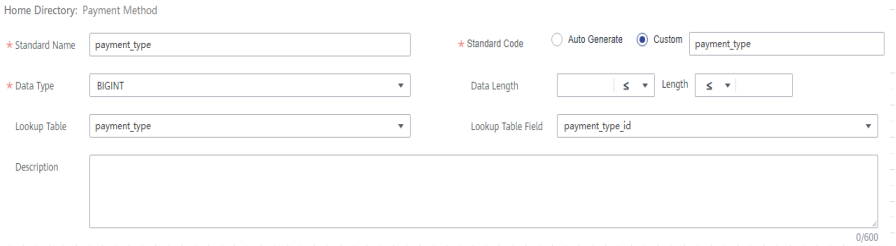


Figure 5-231 Creating a data standard named vendor

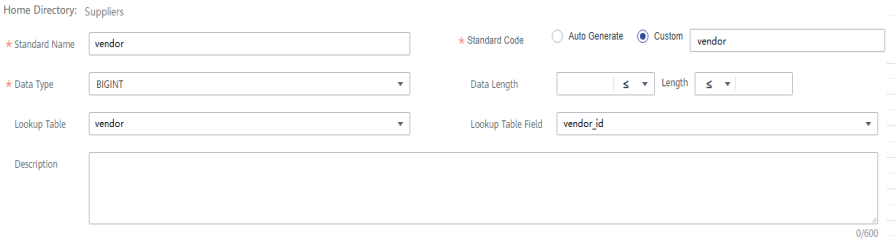
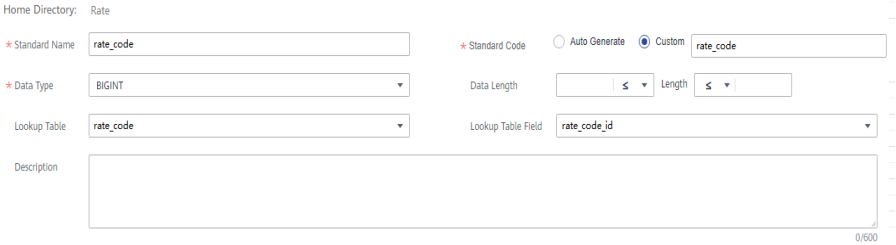


Figure 5-232 Creating a data standard named rate_code



Step 5 Return to the **Data Standards** page, select the three data standards in the list, and click **Publish**.

Step 6 In the **Apply for Publication** dialog box, select a reviewer and click **OK**. Wait for the reviewer to review the application. If you have the reviewer permissions, select **Auto-review** and click **OK**.

----End

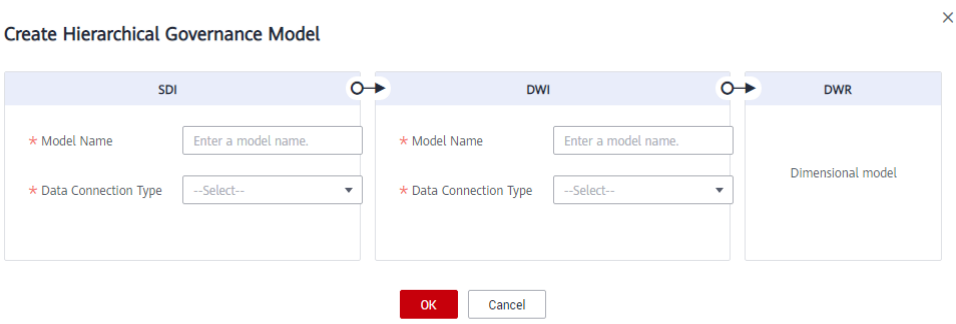
ER Modeling: Creating a Model at the SDI and DWI Layers Respectively

During ER modeling, create an ER model at the SDI and DWI layer, respectively, and import the original data table to the ER model at the SDI layer through by reversing the database, and create a standard service table named **standard travel data** in the ER model at the DWI layer.

Step 1 On the DataArts Architecture page, choose **Models** > **ER Modeling** in the left navigation pane.

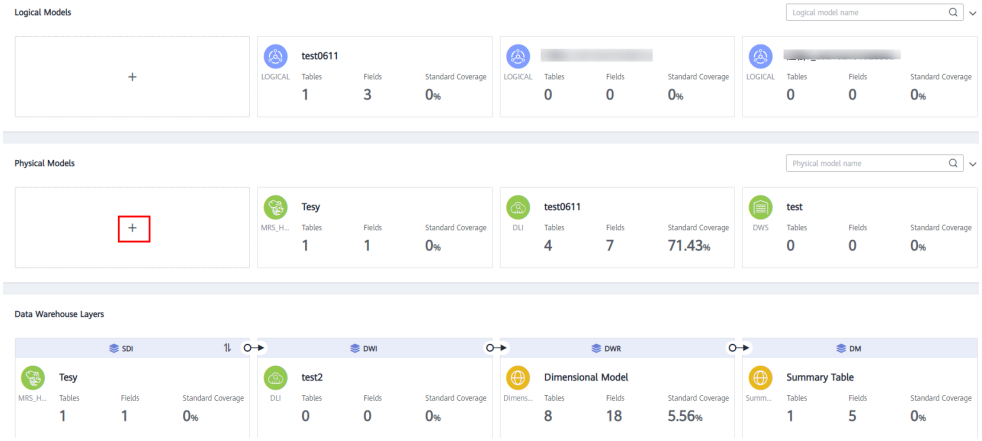
- If no ER model has been created, a dialog box is displayed, asking you to create a hierarchical governance model. You can create an SDI ER model named **sdi** and then create a DWI ER model named **dwi**. Click **OK**.

  Figure 5-233 Dialog box for creating a hierarchical governance model



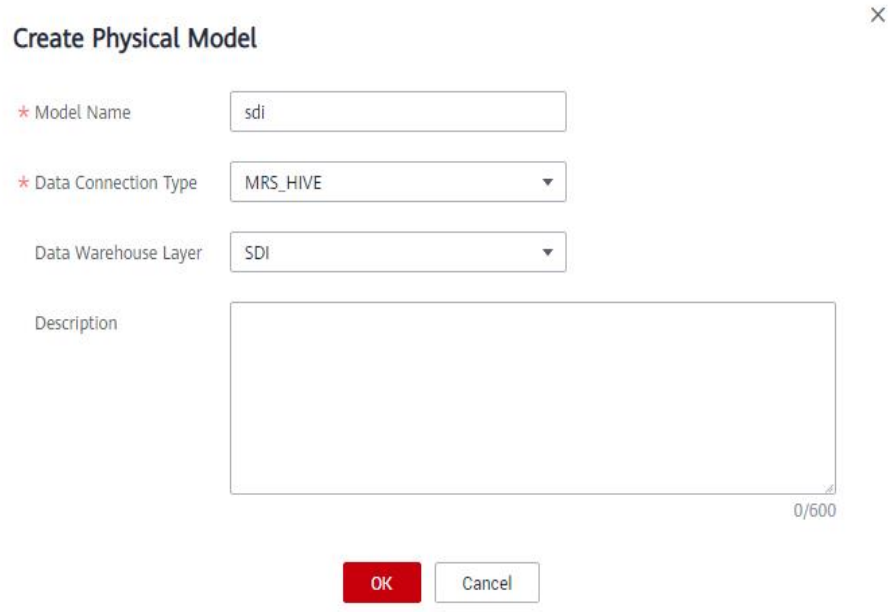
- If you have created ER models before, click + to create physical models, as shown in the following figure.

Figure 5-234 ER Modeling page



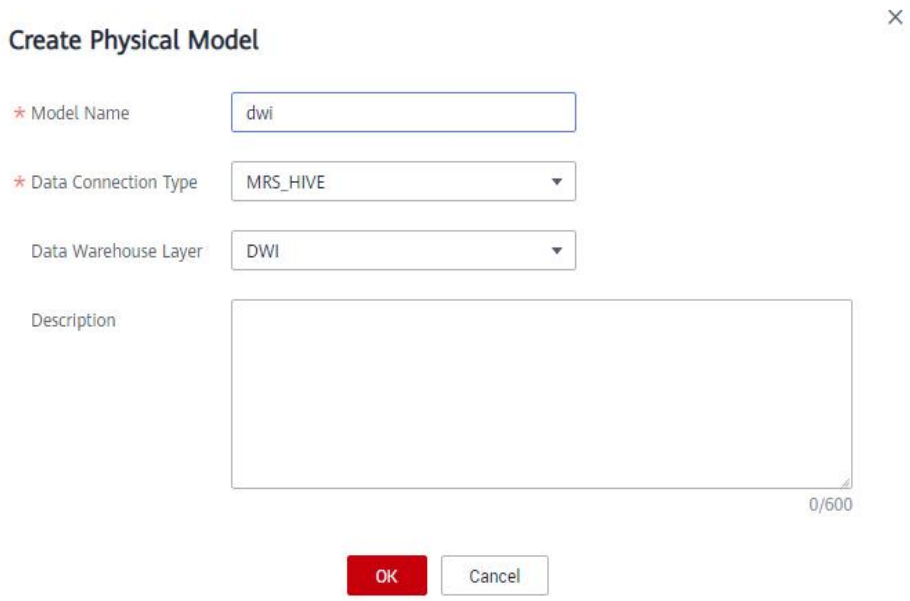
a. Create an ER model at the SDI layer named **sdi**. In the **Physical Models** area, click +. In the displayed dialog box, configure required parameters and click **OK**.

Figure 5-235 Creating a physical model named sdi



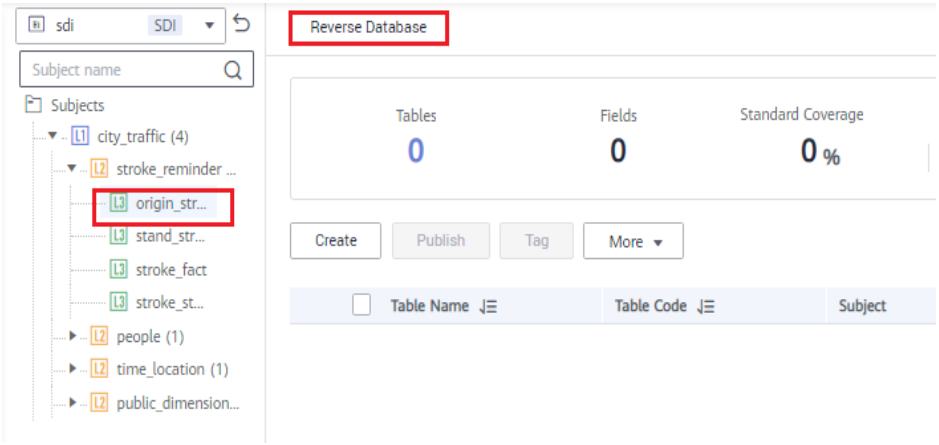
b. Create an ER model at the DWI layer named **dwi**. In the **Physical Models** area, click +. In the displayed dialog box, configure required parameters and click **OK**.

Figure 5-236 Creating a physical model named dwi



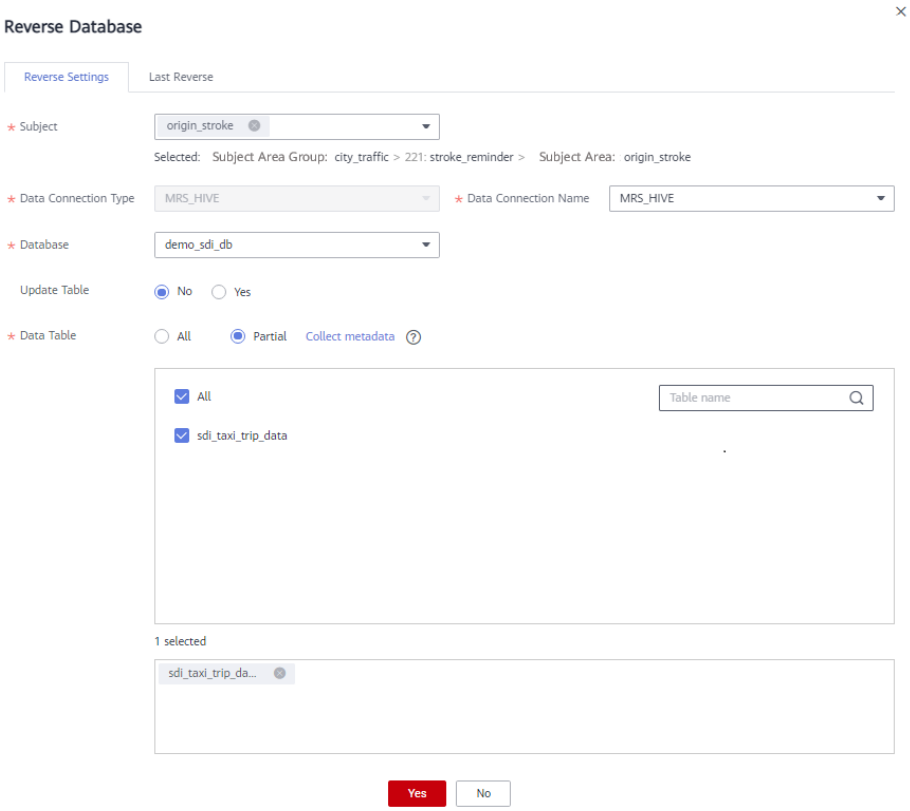
Step 2 In the **Data Warehouse Layers** part, click the newly created SDI ER model. Choose **city_traffic** > **stroke_reminder** > **origin_stroke**, and click **Reverse Database** on the page displayed on the right to import the source table.

Figure 5-237 Model directory



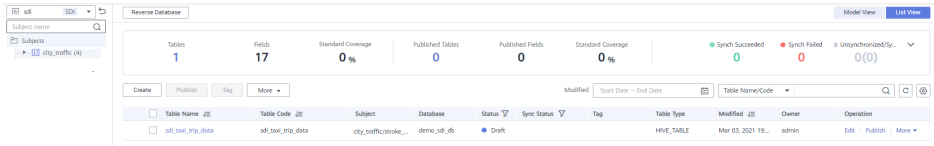
In the **Reverse Database** dialog box, set the parameters and click **OK**. In this example, select the original data table in the source layer database **demo_sdi_db**.

Figure 5-238 Reverse Database dialog box



After the database is reversed successfully, click **Close**. You can view the imported table in the table list.

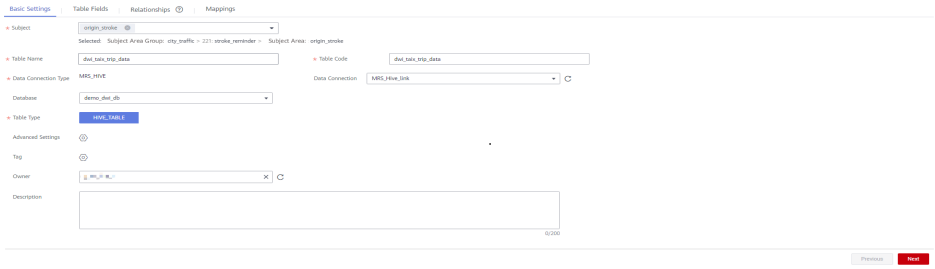
Figure 5-239 Viewing the imported table



Step 3 Perform the following steps to create a standard service table named **standard travel data**:

1. In the **Data Warehouse Layers** part, click the newly created DWI ER model. Choose **city_traffic** > **stroke_reminder** > **origin_stroke**, and click **Create** on the page displayed on the right.
2. On the **Basic Settings** page, set the parameters as follows.

Figure 5-240 Basic settings of the trip data table



3. Click the **Table Fields** tab and then **Add**. Add the fields listed in **Table 5-73**.

   Then click  in the **Data Standard** column of the rows where the vendor ID, rate code ID, and payment type reside to associate with the **Vendor**, **Rate Code ID**, and **Payment Type** standards, respectively. **Figure 5-241** lists the fields to be added.

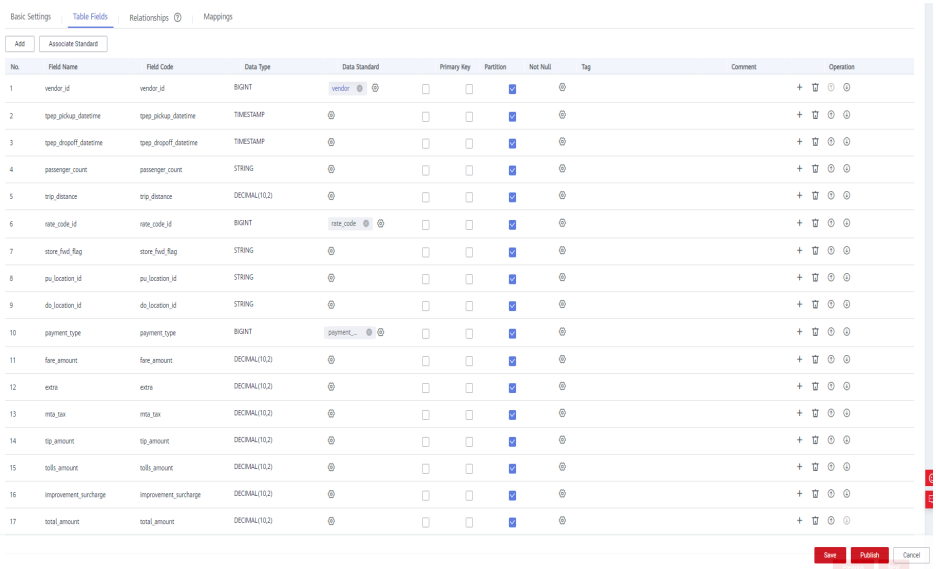
   Table 5-73 Fields in the standard travel data table

No.	Field Name	Field Code	Data Type	Data Standard	Primary Key	Partition	Not Null	Tag
1	Vendor ID	vendor_id	Long integer (BIGINT)	Vendor	Deselected	Deselected	Selected	N/A
2	Pickup Time	tpep_pickup_datetime	Timestamp (TIMESTAMP)	N/A	Deselected	Deselected	Selected	N/A
3	Drop-off Time	tpep_dropoff_datetime	Timestamp (TIMESTAMP)	N/A	Deselected	Deselected	Selected	N/A
4	Passenger Quantity	passenger_count	Character (STRING)	N/A	Deselected	Deselected	Selected	N/A
5	Trip Distance	trip_distance	High-precision (DECIMAL) (10,2)	N/A	Deselected	Deselected	Selected	N/A
6	RateCode	rate_code_id	Long integer (BIGINT)	Rate code	Deselected	Deselected	Selected	N/A
7	Storage Forwarding Flag	store_fwd_flag	Character (STRING)	N/A	Deselected	Deselected	Selected	N/A

No.	Field Name	Field Code	Data Type	Data Standard	Primary Key	Partition	Not Null	Tag
8	Pick up Location	pu_location_id	Character (STRING)	N/A	Deselected	Deselected	Selected	N/A
9	Drop-off Location	do_location_id	Character (STRING)	N/A	Deselected	Deselected	Selected	N/A
10	Payment Type	payment_type	Long integer (BIGINT)	Payment type	Deselected	Deselected	Selected	N/A
11	Fare	fare_amount	High-precision (DECIMAL) (10,2)	N/A	Deselected	Deselected	Selected	N/A
12	Extra Fee	extra	High-precision (DECIMAL) (10,2)	N/A	Deselected	Deselected	Selected	N/A
13	MTA Tax	mta_tax	High-precision (DECIMAL) (10,2)	N/A	Deselected	Deselected	Selected	N/A
14	Handling Fee	tip_amount	High-precision (DECIMAL) (10,2)	N/A	Deselected	Deselected	Selected	N/A
15	Toll	tolls_amount	High-precision (DECIMAL) (10,2)	N/A	Deselected	Deselected	Selected	N/A
16	Improvement Surcharge	improvement_surcharge	High-precision (DECIMAL) (10,2)	N/A	Deselected	Deselected	Selected	N/A

No.	Field Name	Field Code	Data Type	Data Standard	Primary Key	Partition	Not Null	Tag
17	Total Fee	total_amount	High-precision (DECIMAL) (10,2)	N/A	Deselected	Deselected	Selected	N/A

Figure 5-241 Fields in the trip data table



You can perform the following operations on the fields in the standard travel data table

- **Associating with data standards**

  When creating or editing a table, click the **Table Fields** tab. Locate the row that contains a field and click in the **Data Standard** column to associate the field with a data standard. After the field is associated with a data standard and the table is published, a quality job is automatically generated, and a quality rule is generated for each field associated with a data standard. You can monitor the fields based on the data standards and view the field statuses on the **Quality Jobs** page of the DataArts Quality console. For more information about associating data standards, see **Designing Physical Models**.

- **Adding tags**

  Tags are user-defined identifiers. After adding a tag, you can search for related data assets in the DataArts Catalog module with ease.

  When creating or editing a table, click the **Table Fields** tab, locate the row that contains a field, and click in the **Tag** column. In the

displayed dialog box, enter a new tag name and press **Enter** or select an existing tag from the drop-down list.

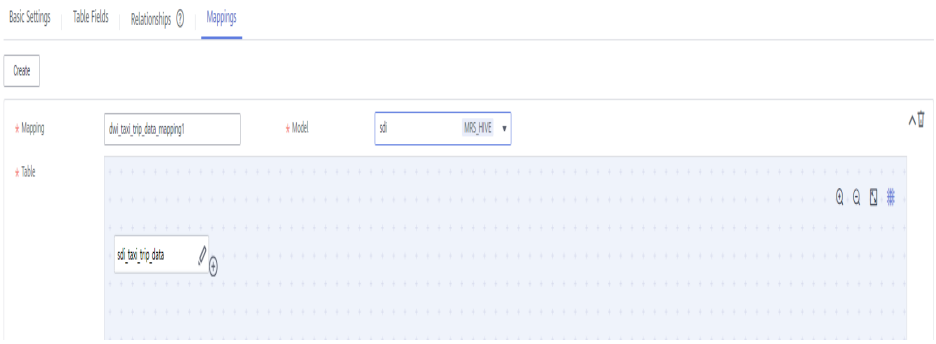
- **Associating with quality rules**

  After creating a table, you can associate fields in the table with quality rules. After the association is complete and the table is published, a quality job is automatically created on the **DataArts Quality** page after the table is published. If the table has been published, the system automatically updates the quality job. For more information about associating quality rules, see **Associating with Quality Rules**.

4. Click the **Mappings** tab and create mappings to design data sources of the table.
   - If the table field comes from different relationship models, you must create multiple mappings. In each mapping, you only need to set the source field for the field that comes from the current mapping.
   - If the table fields come from multiple tables in the same ER model, you can create a mapping. You can set **Join** for multiple tables of the mapping and set source fields for the fields in the table.

   In this example, you only need to create one mapping. Click **Create** to create a mapping.
   - **Mapping** is automatically generated, but is also configurable.
   - **Model**: Select **sdi**.
   - **Table**: Select the original data table **sdi_taxi_trip_data**, from where data of the standard travel table comes.

     Figure 5-242 Creating a mapping

   - **Field Mapping**

     In the **Field Mapping** area, set source fields for the fields in the table in sequence. The selected source fields must have the same meaning as the fields in the table. As shown in **Figure 5-243**, the generated SQL statement is displayed at the bottom of the **Field Mapping** area.

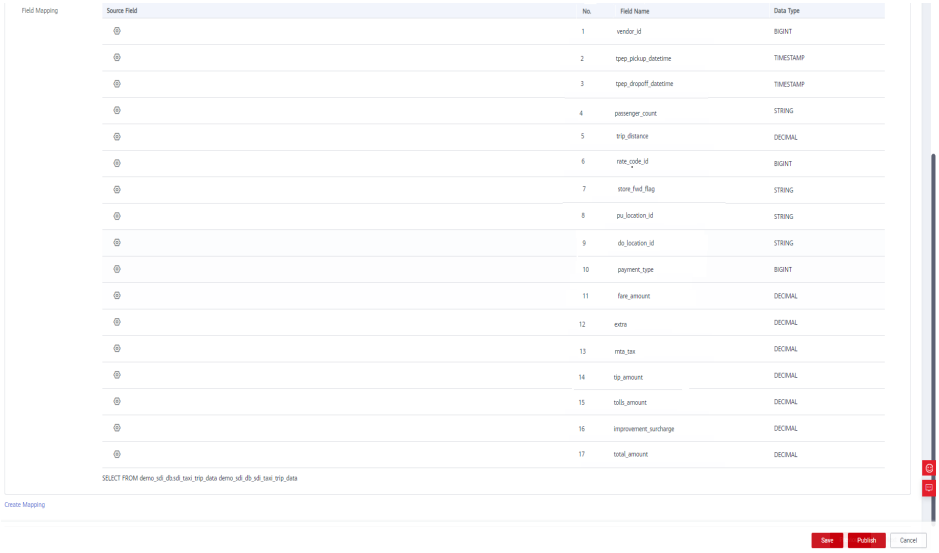
NOTE

- On the DataArts Architecture page, choose **Metrics** > **Configuration Center** in the left navigation pane, and click the **Functions** tab. On the page displayed, if **Create data development jobs** is selected for **Model Design Process**, the system creates an ETL job during data development based on the table mapping information during table release. An ETL node is generated for each mapping, and the job name starts with *Database name_Table code*. Currently, this function is in the internal test stage. Only DLI-to-DLI and DLI-to-DWS mapping jobs can be created.

  You can choose **DataArts Factory** > **Job Development** to view the created ETL jobs. By default, ETL jobs are scheduled at 00:00 every day.

- In this example, the function of automatically creating ETL jobs is not enabled. The function provides only the data flow direction for data development. During data development, you can refer to the mapping to write SQL scripts.

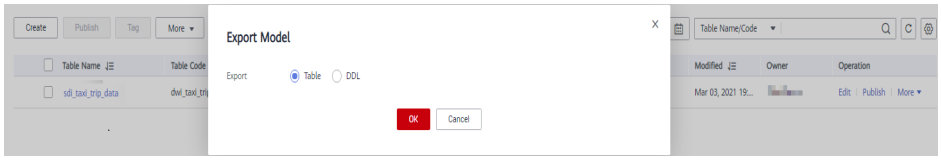
Figure 5-243 Field Mapping



5. After configuring the mapping, you have finished configuring the taxi trip data table. Click **Save**.

Step 4 Select the created model and choose **More** > **Export**. In the displayed dialog box, select **Table** for **Export** and click **OK** to export the model. Then export the **sdi** model in the same way. The exported models can be used as backups and imported when needed in the future.

Figure 5-244 Export Model dialog box



Step 5 Publish table models.

1. Publish the source table imported to the SDI ER model in **Step 2**. After the table is published, you can use DataArts Studio to manage and monitor the source table.

Return to the **ER Modeling** page, select the **sdi** model in the model directory. Select the **sdi_taxi_trip_data** table in the list on the right, and click **Publish**. In the dialog box displayed, select a reviewer and click **OK**. Wait for the reviewer to review the application. If you have the reviewer permissions, select **Auto-review** and click **OK**.

2. Publish a table of the DWI ER model.

   Return to the **ER Modeling** page, select the **dwi** model in the model directory. Select the **dwi_table_trip_data** table in the list on the right, and click **Publish**. In the dialog box displayed, select a reviewer and click **OK**. Wait for the reviewer to review the application. If you have the reviewer permissions, select **Auto-review** and click **OK**.

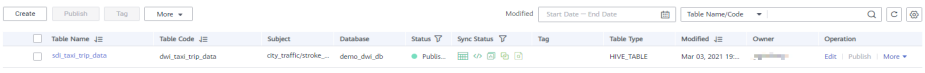
Step 6 After the application is approved, you can view **Status** and **Sync Status** on the **ER Modeling** page.

Publishing is an asynchronous operation. You can click  to refresh the status. After table publishing application is approved, the system performs operations such as creating tables and synchronizing technical assets and business assets based on the configurations of **Model Design Process** on the **Function Settings** tab page in **Configuration Center**. The synchronization status is displayed in the **Sync Status** column of the table on the **Information Architecture** page.

- If the **Sync Status** is successful, the table is published successfully. Move the cursor over  in the **Sync Status** column. If the message **Table created successfully** is displayed, the table has been successfully created in the corresponding data source.
- If one or more items in the in the **Sync Status** column fail to be synchronized, you can refresh the status. If the fault persists, choose **More** > **View History** to view logs.

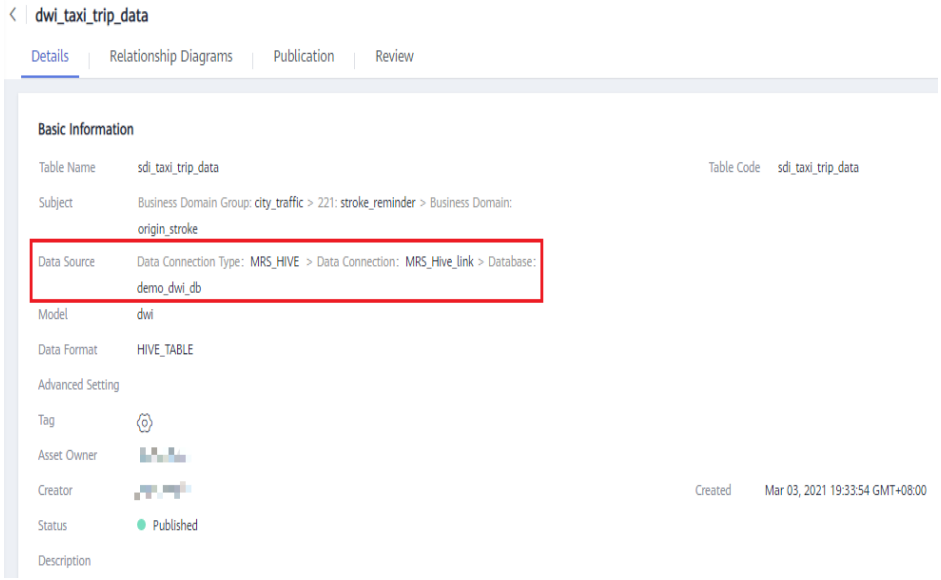
  Locate the failure cause based on the error log and rectify the fault. Then return to the **ER Modeling** page, select the tables to be synchronized from the list, and choose **More** > **Synchronize** to synchronize the tables again. If the fault persists, contact technical support.

Figure 5-245 Checking the table status



Click a table name in the list to view the table details. **Data Source** indicates the location of the table.

Figure 5-246 Table details



----End

Creating and Publishing Dimensions for the DWR Layer

During dimension modeling, create three lookup table dimensions (**vendor**, **rate_code**, and **payment_type**) and one hierarchy dimension (**date**) for the DWR layer.

Step 1 On the DataArts Architecture console, choose **Models** > **Dimensional Modeling** in the navigation pane on the left.

Step 2 Create the three lookup table dimensions listed in **Table 5-74**.

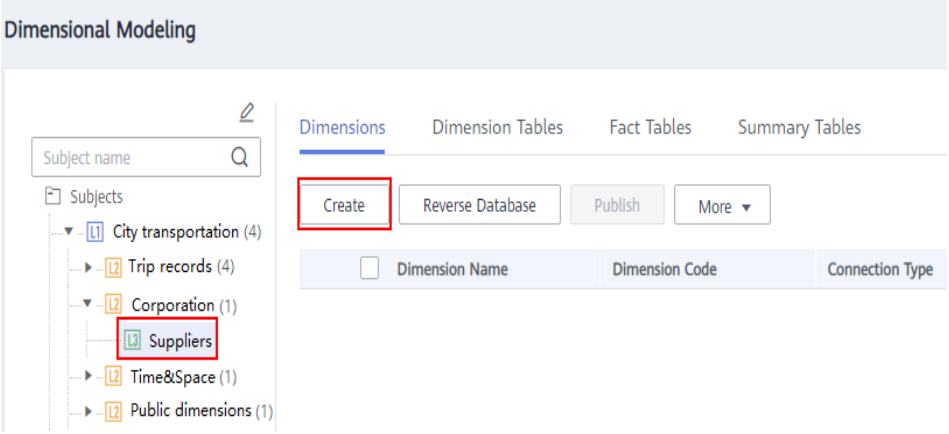
Table 5-74 Lookup table dimensions

*Subject	*Dimension Name	* Dimension Code	*Type	*Owner	Description	*Data Connection Type	*Data Connection Name	*Database	Lookup Table
vendor	dim_vendor	dim_vendor	Lookup table	-	None	MRS_HIVE	mrs_hive_link	demo_dwr_db	vendor
public_dimension	dim_rate_code	dim_rate_code	Lookup table	-	None	MRS_HIVE	mrs_hive_link	demo_dwr_db	rate

*Subject	*Dimension Name	* Dimension Code	*Type	*Owner	Description	*Data Connection Type	*Data Connection Name	*Database	Lookup Table
public_dimension	dim_payment_type	dim_payment_type	Lookup table	-	None	MRS_HIVE	mrs_hive_link	demo_dwr_db	payment_type

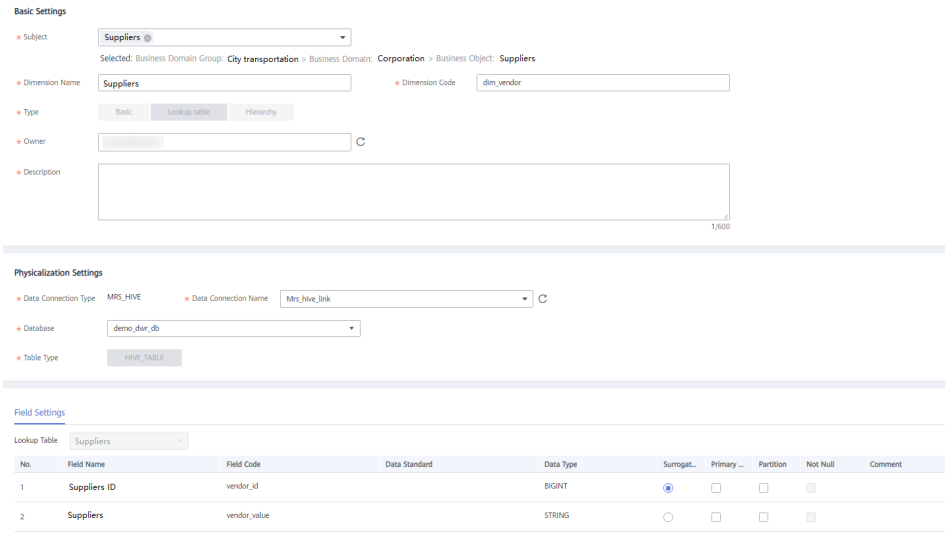
1. Click the **Dimensions** tab, choose **City transportation** > **Corporation** > **Suppliers** in the subject tree, and click **Create** to create a dimension named **dim_vendor**.

   Figure 5-247 Dimensional modeling

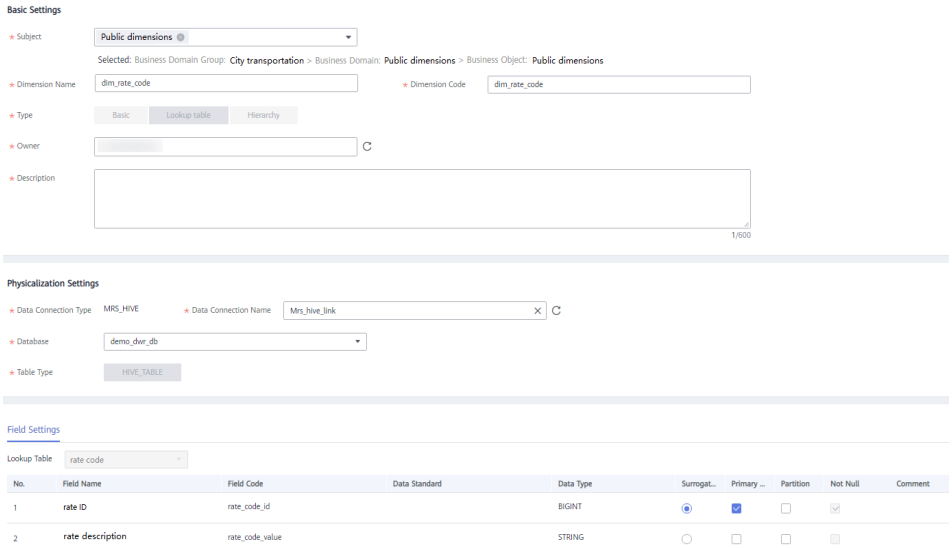
2. On the **Create Dimension** page, set the parameters as shown in the figure below and click **Save**.

Figure 5-248 Creating a dimension named dim_vendor



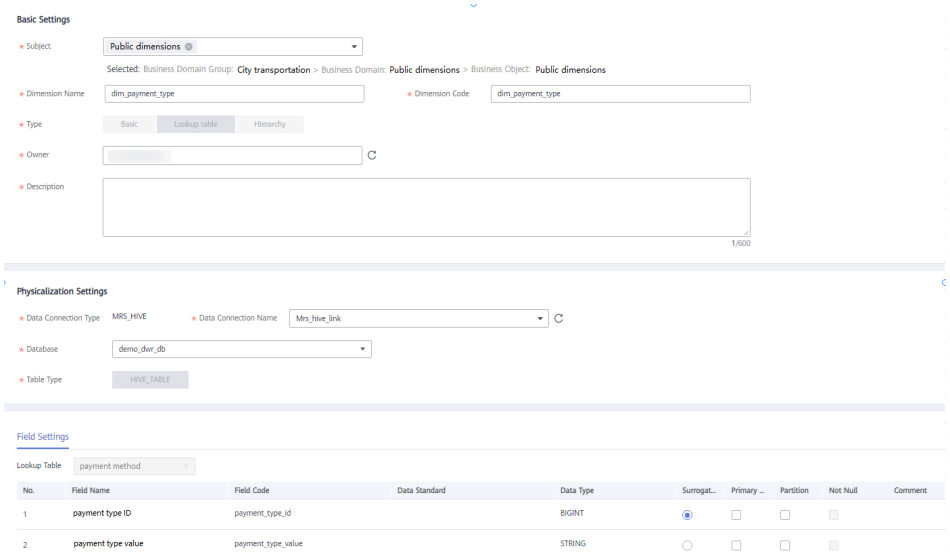
3. Click the **Dimensions** tab, choose **City transportation** > **Public dimensions** > **Public dimensions** in the subject tree, and click **Create** to create a dimension named **dim_rate_code**. On the **Create Dimension** page, set the parameters as shown in the figure below and click **Save**.

Figure 5-249 Creating a dimension named dim_rate_code



4. Click the **Dimensions** tab, choose **City transportation** > **Public dimensions** > **Public dimensions** in the subject tree, and click **Create** to create a dimension named **dim_payment_type**. On the **Create Dimension** page, set the parameters as shown in the figure below and click **Save**.

Figure 5-250 Creating a dimension named dim_payment_type



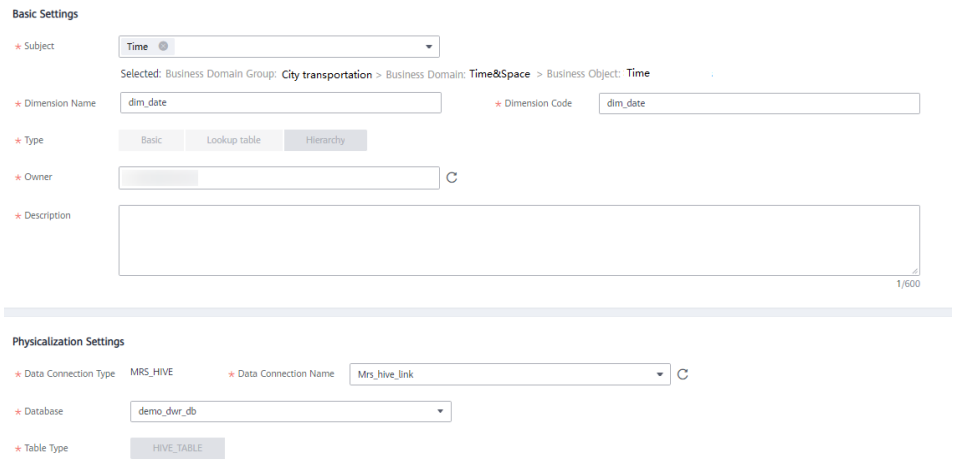
Step 3 Create a hierarchy dimension named **dim_date**.

1. On the **Dimensional Modeling** tab page, choose **City transportation** > **Time&Space** > **Time** in the subject tree. Then click **Create** on the **Dimensions** tab page to create a dimension named **dim_date**.
2. Configure the basic settings and physicalization settings as shown in the figure below.

Table 5-75 Date dimension

*Subject	*Dimension Name	* Dimension English Name	*Type	*Owner	Description	*Data Connection Type	*Data Connection Name	*Database
date	dim_date	dim_date	Hierarchy	-	None	MRS_HIVE	mrs_hive_link	demo_dwr_db

Figure 5-251 Date dimension



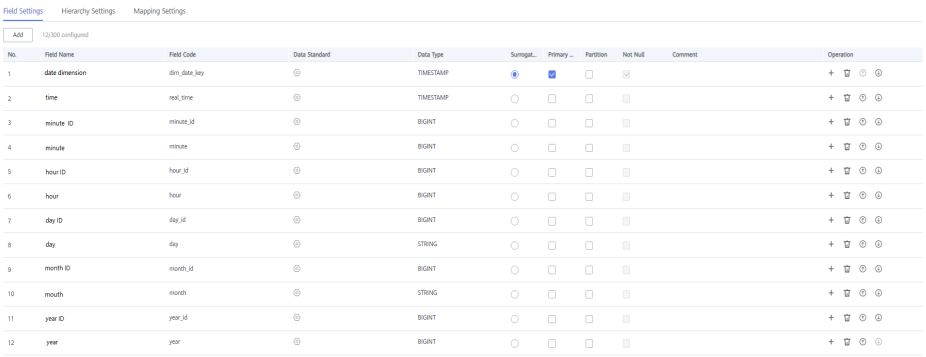
3. In the **Field Settings** area, add fields as described in the table below.

Table 5-76 Field settings

No.	Field Name	Field Code	Data Standard	Data Type	Surrogate Key	Primary Key	Partition	Not Null
1	dim_date_key	dim_date_key	-	TIMESTAMP	Selected	Selected	Not selected	Selected
2	real_time	real_time	-	TIMESTAMP	Not selected	Not selected	Not selected	Not selected
3	minute_id	minute_id	-	BIGINT	Not selected	Not selected	Not selected	Not selected
4	minute	minute	-	BIGINT	Not selected	Not selected	Not selected	Not selected
5	hour_id	hour_id	-	BIGINT	Not selected	Not selected	Not selected	Not selected
6	hour	hour	-	BIGINT	Not selected	Not selected	Not selected	Not selected
7	day_id	day_id	-	BIGINT	Not selected	Not selected	Not selected	Not selected
8	day	day	-	STRING	Not selected	Not selected	Not selected	Not selected

No.	Field Name	Field Code	Data Standard	Data Type	Surrogate Key	Primary Key	Partition	Not Null
9	month_id	month_id	-	BIGINT	Not selected	Not selected	Not selected	Not selected
10	month	month	-	STRING	Not selected	Not selected	Not selected	Not selected
11	year_id	year_id	-	BIGINT	Not selected	Not selected	Not selected	Not selected
12	year	year	-	BIGINT	Not selected	Not selected	Not selected	Not selected

Figure 5-252 Field settings



4. In the **Hierarchy Settings** area, click **Add** to create two layers as shown in the figures below.

Figure 5-253 Layer 1

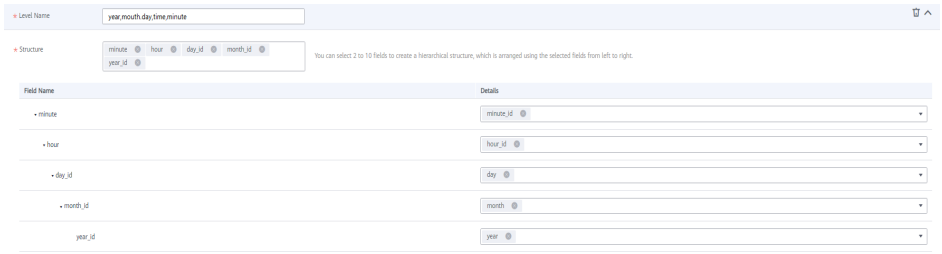
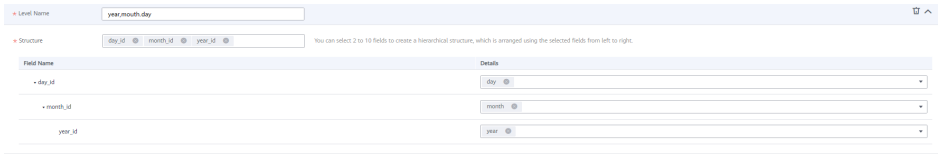


Figure 5-254 Layer 2



5. Click **Save**.

Step 4 Return to the **Dimensions** tab page, select the four new dimensions in the dimension list, and click **Publish**.

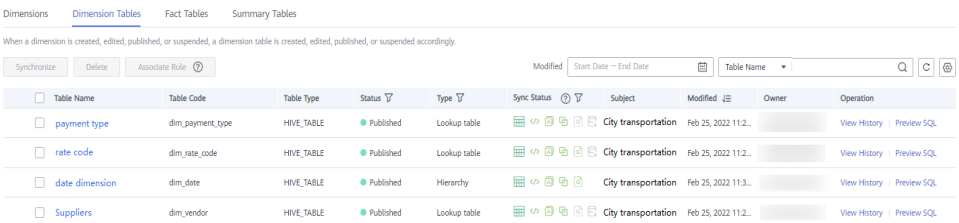
Step 5 In the **Apply for Publication** dialog box, select a reviewer and click **OK**. Wait for the reviewer to review the application. If you have the reviewer permissions, select **Auto-review** and click **OK**.

Step 6 After a dimension is published and approved, the system automatically creates a dimension table for the dimension. The name and code of the dimension table are the same as those of the dimension. On the **Dimensional Modeling** page, click the **Dimension Tables** tab to view the created dimension table.

In the dimension table list, you can view **Sync Status** of the dimension tables.

- If all items in **Sync Status** are displayed as **Succeeded**, the dimension is published and the dimension table is created in the database.
- If an item in **Sync Status** is displayed as **Failed**, click **View History** in the row. On the page displayed, click the **History** tab to view logs. Troubleshoot the fault based on the logs. After the fault is rectified, select the dimension table, click **Synchronize** above the dimension table list, and click **OK** in the dialog box displayed. If the fault persists, contact technical support for assistance.

Figure 5-255 Sync Status of the dimension tables



----End

Creating and Publishing a Fact Table for the DWR Layer

During dimensional modeling, create a fact table named **stroke_order** for the DWR layer.

Step 1 On the DataArts Architecture console, choose **Models** > **Dimensional Modeling** in the navigation pane on the left.

Step 2 Click the **Fact Tables** tab, choose **City transportation** > **Trip records** > **Trip facts** in the subject tree, and click **Create** to create a fact table named **stroke_order**.

In the **Basic Settings** area on the **Create Fact Table** page, set the following parameters:

- **Subject**: **Subject Area Group: City transportation > Subject Area: Trip records > Business Object: Trip facts**
- **Table Name**: **stroke_order**
- **Table English Name**: **fact_stroke_order**
- **Data Connection Type**: **MRS_HIVE**
- **Data Connection Name**: **mrs_hive_link**
- **Database**: **demo_dwr_db**
- **Table Type**: **HIVE_TABLE**
- **Owner**: an owner in the drop-down list box
- **Description**: None

In the **Field Settings** area, choose **Create** > **Dimension**. In the dialog box displayed, select the dimensions **rate_code**, **vendor**, **payment_type**, and **date**, and click **OK**. Choose **Create** > **Dimension**. In the dialog box displayed, select the dimension **date** and click **OK**. In the dimension field list, adjust the sequence of the dimension fields and modify the information about the two **date** dimensions, as listed in **Table 5-77**.

Table 5-77 Dimension fields

No.	Field Name	Field Code	Data Type	Primary Key	Partition	Not Null	Associated Standard	Associated Dimension	Role	Description
1	rate_code_id	rate_code_id	BIGINT	Not selected	Not selected	Not selected	-	rate_code	dim_	-
2	vendor_id	vendor_id	BIGINT	Not selected	Not selected	Not selected	-	vendor	dim_	-

No.	Field Name	Field Code	Data Type	Primary Key	Partition	Not Null	Associated Standard	Associated Dimension	Role	Description
3	payment_type_id	payment_type_id	BIGINT	Not selected	Not selected	Not selected	-	payment_type	dim_	-
4	pickup_date_key	dim_pickup_date_key	TIMESTAMP	Not selected	Not selected	Not selected	-	Date	dim_pickup	Date dimension table
5	tpep_dropoff_datetime	dim_dropoff_date_key	TIMESTAMP	Not selected	Not selected	Not selected	-	Date	dim_dropoff	Date dimension table

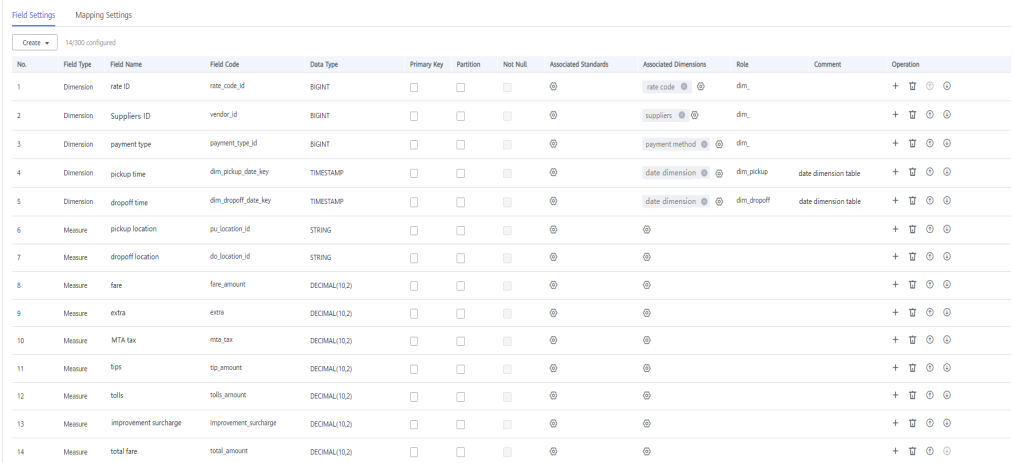
In the **Field Settings** area, choose **Create** > **Measure** and create the fields listed in **Table 5-78** in sequence.

Table 5-78 Measure fields

No.	Field Name	Field Code	Data Type	Primary Key	Partition	Not Null	Associated Standard
6	pu_location_id	pu_location_id	STRING	Not selected	Not selected	Not selected	-
7	do_location_id	do_location_id	STRING	Not selected	Not selected	Not selected	-
8	fare_amount	fare_amount	DECIMAL (10,2)	Not selected	Not selected	Not selected	-

No.	Field Name	Field Code	Data Type	Primary Key	Partition	Not Null	Associated Standard
9	extra	extra	DECIMAL (10,2)	Not selected	Not selected	Not selected	-
10	mta_tax	mta_tax	DECIMAL (10,2)	Not selected	Not selected	Not selected	-
11	tip_amount	tip_amount	DECIMAL (10,2)	Not selected	Not selected	Not selected	-
12	tolls_amount	tolls_amount	DECIMAL (10,2)	Not selected	Not selected	Not selected	-
13	improvement_surcharge	improvement_surcharge	DECIMAL (10,2)	Not selected	Not selected	Not selected	-
14	total_amount	total_amount	DECIMAL (10,2)	Not selected	Not selected	Not selected	-

Figure 5-256 Fact table fields



Step 3 After the configuration, click **Publish**.

Step 4 In the dialog box displayed, select a reviewer and click **OK**. Wait for the reviewer to review the application. If you have the reviewer permissions, select **Auto-review** and click **OK**.

Step 5 Return to the **Fact Tables** tab page, find the new fact table in the list, and view **Sync Status**.

- If all items in **Sync Status** are displayed as **Succeeded**, the fact table is published and created in the database.
- If an item in **Sync Status** is displayed as **Failed**, choose **More** > **View History**. On the page displayed, click the **History** tab to view logs. Troubleshoot the fault based on the logs. After the fault is rectified, choose **More** > **Synchronize** above the fact table list, and click **OK** in the dialog box displayed. If the fault persists, contact technical support for assistance.

----End

Creating and Publishing Technical Metrics

In this example, you need to create the technical metrics listed in **Table 5-79** and **Table 5-80**.

Table 5-79 Atomic metrics

*Metric Name	* Metric Code	Data Table	*Subject	*Expression	Description
sum_total_amount	sum_total_amount	Itinerary order	stroke_fact	sum (total amount)	None

Table 5-80 Derivative metrics

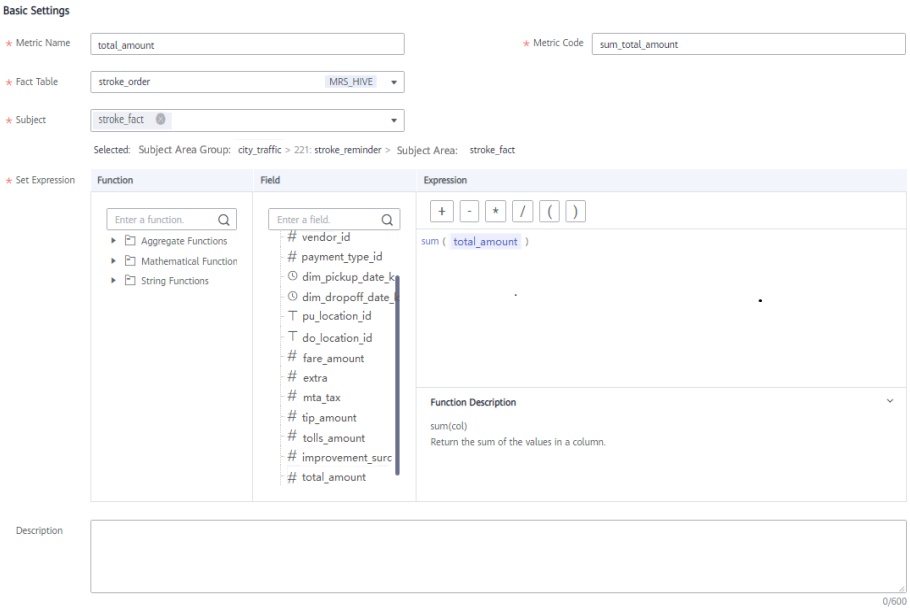
Metric	*Data Table	*Subject	*Atomic Metric	Statistical Dimension	Time Filter	General Filter
total_amount_(payment_type)	Itinerary order	stroke_statistic	total_amount	payment_type	None	None
total_amount_(rate_code)	Itinerary order	stroke_statistic	total_amount	rate_code	None	None
total_amount_(vendor,stroke_order.dim_dropoff_date_key)	Itinerary order	stroke_statistic	total_amount	**vendor** and **stroke_order.dim_dropoff_date_key**	None	None

Step 1 On the DataArts Architecture console, choose **Metrics** > **Technical Metrics** in the navigation pane on the left.

Step 2 Create an atomic metric named **total_amount** to collect statistics on fares.

1. Click the **Atomic Metrics** tab and click **Create**.
2. On the **Create Atomic Metric** page, set the parameters as shown in the figure below and click **Publish**.

Figure 5-257 Creating an atomic metric



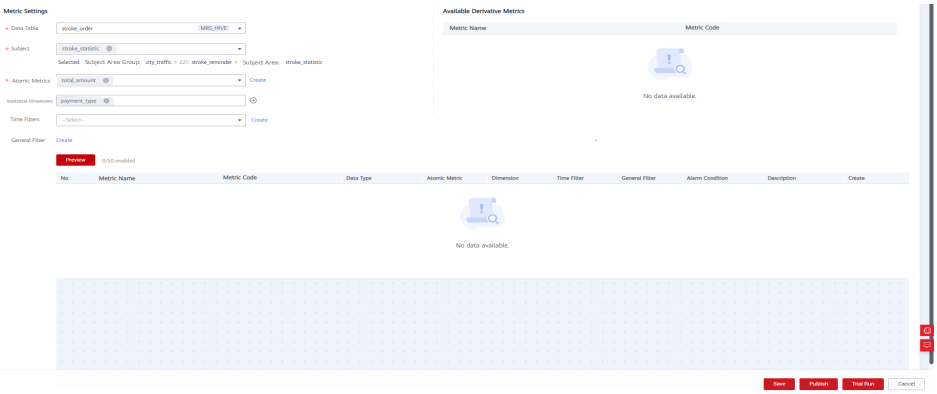
3. Wait for the reviewer to review the application. After the application is approved, the atomic metric will be created.

Step 3 Create three derivative metrics.

- Create **total_amount_(payment_type)** to collect statistics on the total fares based on **payment_type**.

  On the **Technical Metrics** page, click the **Derivative Metrics** tab and click **Create**. On the **Create Derivative Metric** page, set the parameters as shown in the figure below. After the configuration is complete, click **Trial Run**. In the dialog box displayed, click **Execute**. If the trial running is successful, click **Save**.

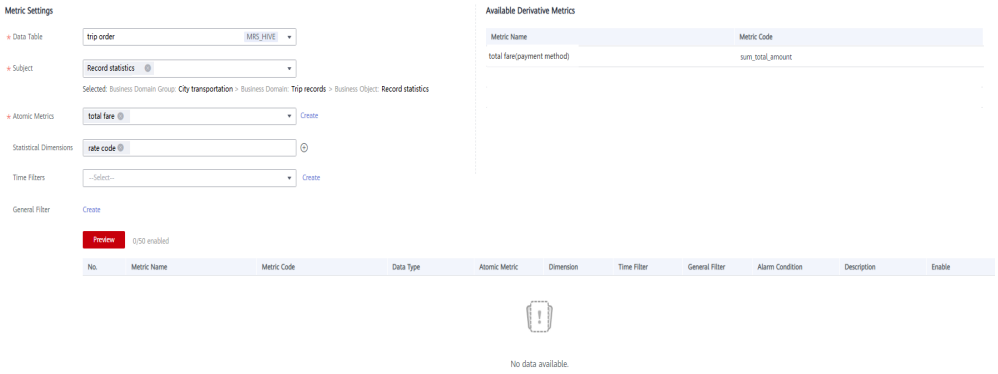
Figure 5-258 Creating a derivative metric named total_amount_(payment_type)



- Create **total_amount_(rate_code)** to collect statistics on the total fares based on **rate_code**.

  On the **Technical Metrics** page, click the **Derivative Metrics** tab and click **Create**. On the **Create Derivative Metric** page, set the parameters as shown in the figure below. After the configuration is complete, click **Trial Run**. In the dialog box displayed, click **Execute**. If the trial running is successful, click **Save**.

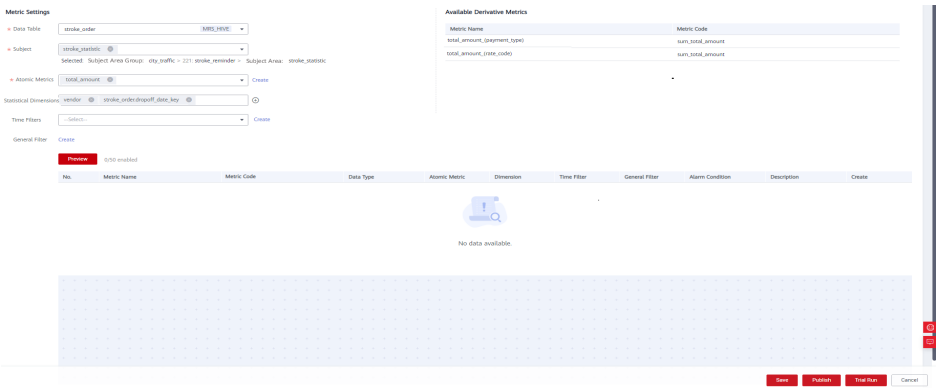
  Figure 5-259 Creating a derivative metric named total_amount_(rate_code)

- Create **total_amount_(vendor,stroke_order.dim_dropoff_date_key)** to collect statistics on the total fares based on **vendor**.

  On the **Technical Metrics** page, click the **Derivative Metrics** tab and click **Create**. On the **Create Derivative Metric** page, set the parameters as shown in the figure below. After the configuration is complete, click **Trial Run**. In the dialog box displayed, click **Execute**. If the trial running is successful, click **Save**.

  Figure 5-260 Creating a derivative metric named total_amount_(vendor,stroke_order.dim_dropoff_date_key)

Step 4 Return to the **Derivative Metrics** tab page, select the three derivative metrics and click **Publish**. In the dialog box displayed, select a reviewer and click **OK**. Wait for the reviewer to review the application. If you have the reviewer permissions, select **Auto-review** and click **OK**.

----End

Creating and Publish Summary Tables for the DM Layer

Create the three summary tables listed in **Table 5-81** for the DM layer.

Table 5-81 Summary tables

*Subject	*Table Name	* Table English Name	Statistical Dimension	Data Connection Type	*Data Connection Name	*Database	Owner	Description
stroke_statistic	dws_payment_type	dws_payment_type	payment_type	MRS_HIVE	mrs_hive_link	demo_dm_db	-	None
stroke_statistic	dws_rate_code	dws_rate_code	rate_code	MRS_HIVE	mrs_hive_link	demo_dm_db	-	None
stroke_statistics	dws_vendor	dws_vendor	**vendor** and **stroke_order.dim_dropoff_date_key**	MRS_HIVE	mrs_hive_link	demo_dm_db	-	None

Step 1 On the DataArts Architecture console, choose **Models** > **Dimensional Modeling** in the left navigation pane.

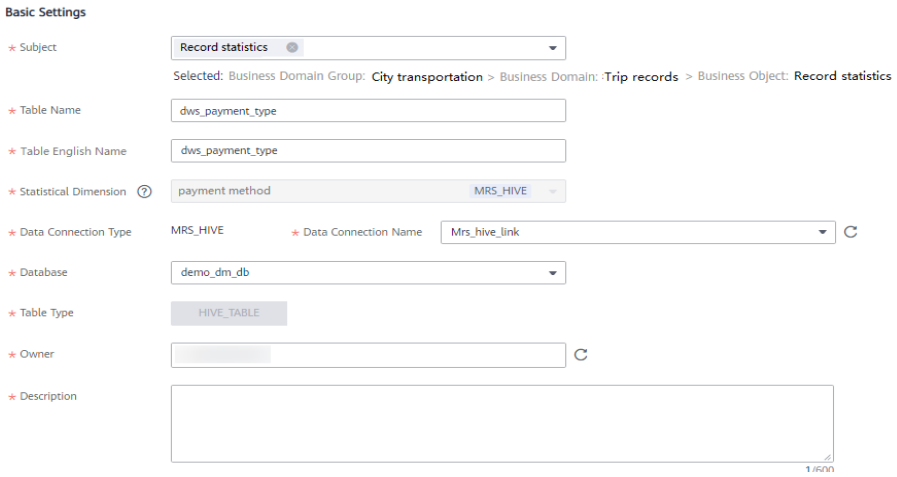
Step 2 Click the **Summary Tables** tab.

Step 3 Create three summary tables: **payment_type**, **rate_code**, and **vendor**.

1. On the **Summary Tables** page, choose **City transportation** > **Trip records** > **Record statistics** in the directory tree, and click **Create** to create a summary table named **dws_payment_type**. On the **Create Summary Table** page, set the parameters and click **Save**.

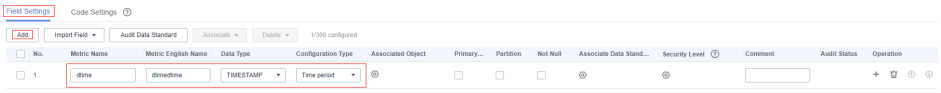
   Set the basic settings as shown in the figure below.

Figure 5-261 Creating a summary table named dws_payment_type



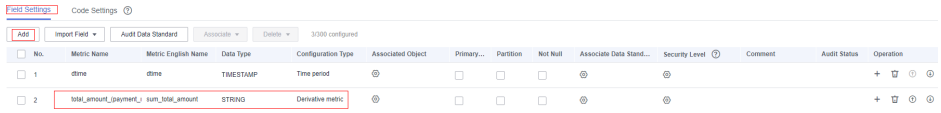
On the **Field Settings** tab page, click **Add**. enter the time field name, and select the data type.

Figure 5-262 Field settings



On the **Field Settings** tab page, click **Add** to add the derivative metric **total_amount_(payment_mode)**. You can add only published derivative or compound metrics that are associated with the specified statistical dimension.

Figure 5-263 Field settings



Click **Save**.

2. On the **Summary Tables** page, choose **City transportation** > **Trip records** > **Record statistics** in the directory tree, and click **Create** to create a summary table named **dws_rate_code**. On the **Create Summary Table** page, set the parameters and click **Save**.

Figure 5-264 Creating a summary table named dws_rate_code (Basic Settings)

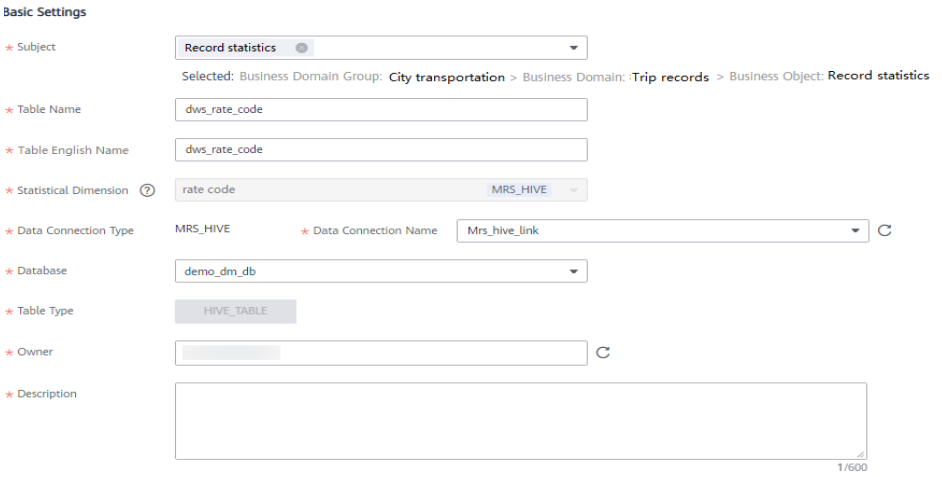
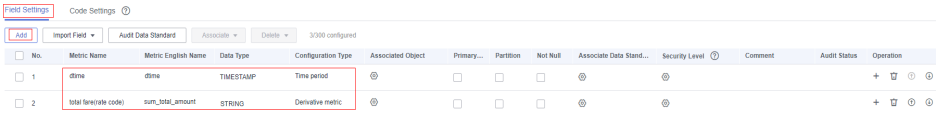


Figure 5-265 Creating a summary table named dws_rate_code (Field Settings)



3. On the **Summary Tables** page, choose **City transportation** > **Trip records** > **Record statistics** in the directory tree, and click **Create** to create a summary table named **dws_vendor**. On the **Create Summary Table** page, set the parameters and click **Save**.

Figure 5-266 Creating a summary table named dws_vendor (Basic Settings)

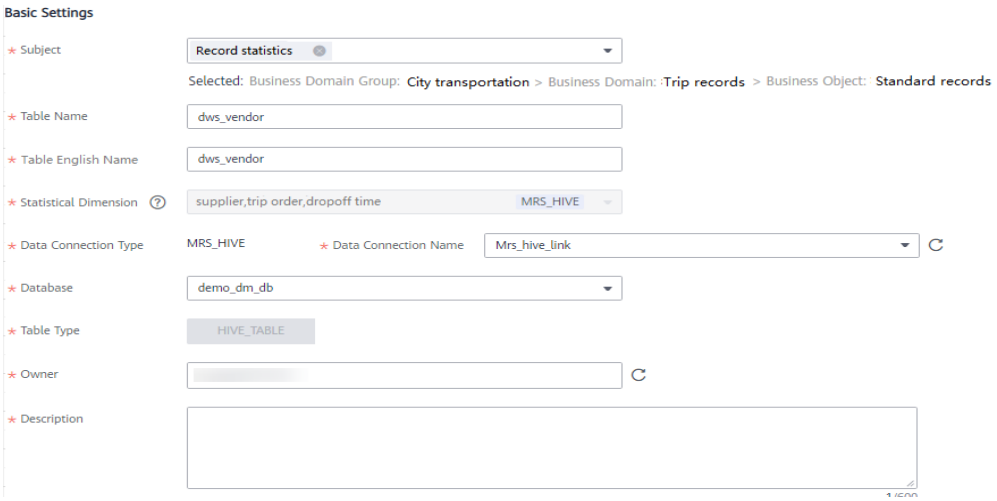
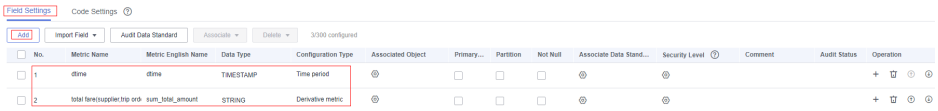


Figure 5-267 Creating a summary table named dws_vendor (Field Settings)



Step 4 Return to the **Dimensions** tab page, select the three new summary tables in the dimension list, and click **Publish**.

Step 5 In the dialog box displayed, select a reviewer and click **OK**. After the reviewer approves the publishing application, the summary table is automatically created. If you have the reviewer permissions, select **Auto-review** and click **OK**.

Step 6 Return to the **Summary Tables** tab page, find the new summary tables in the list, and view **Sync Status**.

- If all items in **Sync Status** are displayed as **Succeeded**, the summary tables are published and created in the database.
- If an item in **Sync Status** is displayed as **Failed**, choose **More** > **View History** in the row. On the page displayed, click the **History** tab to view logs. Troubleshoot the fault based on the logs. After the fault is rectified, choose **More** > **Synchronize** above the summary table list, and click **OK** in the dialog box displayed. If the fault persists, contact technical support for assistance.

----End

6 DataArts Factory

6.1 Overview

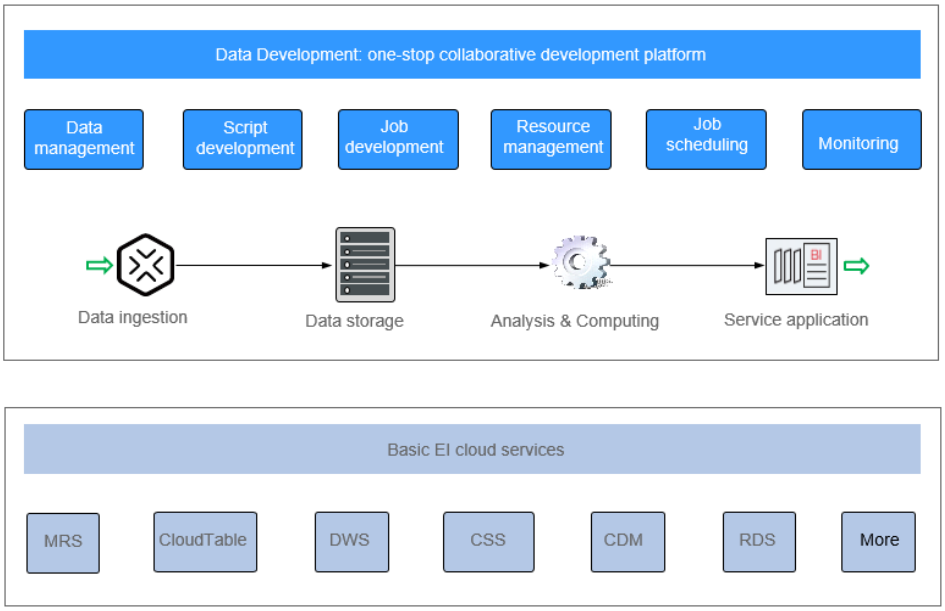
DataArts Factory is a one-stop big data collaborative development platform that provides fully managed big data scheduling capabilities. It manages various big data services, making big data more accessible than ever before and helping you effortlessly build big data processing centers.

DataArts Factory used to be Data Lake Factory (DLF). Therefore, in this document, both Data Lake Factory and DLF can be used to refer to DataArts Factory.

Introduction to DataArts Factory

DataArts Factory enables a variety of operations such as data management, script development, job development, job scheduling, and monitoring, facilitating data analysis and processing.

Figure 6-1 DataArts Factory architecture



Main Functions

Table 6-1 Main functions of DataArts Factory

Function	Description
Data management	• Manages multiple data warehouses, such as GaussDB(DWS), DLI and MRS Hive. • Manages data tables using the GUI or data definition language (DDL).
Script development	• Provides an online script editor that allows more than one operator to collaboratively develop and debug SQL, Python, and Shell scripts online. • Allows use of variables and functions.
Job development	• Provides a graphical designer that allows you to quickly build a data processing workflow by drag-and-drop. • Presets multiple task types such as data integration, SQL, and Shell, and completes data analysis and processing by dependency between tasks. • Supports job import and export.
Resource management	Supports unified management of file, jar, and archive resources used during script and job development.
Job scheduling	Schedules jobs to run once or recursively and use events to trigger scheduling jobs. If the scheduling frequency is set to hour, the scheduling period can be based on interval hour or discrete hour.
Monitoring	• You can run, suspend, restore, or terminate a job. • You can view the operation details of each job and each node in the job. • You can use various methods to receive notifications when a job or task error occurs.

Objects in DataArts Factory

- Data connection: A data connection is a collection of information required for accessing data storage (computing) space, including the connection type, name, and login information.
- Solution: A solution provides users with convenient and systematic management operations to better meet service requirements and objectives. Each solution can contain one or more business-related jobs, and one job can be used by multiple solutions.
- Job: A job is composed of one or more nodes and can be executed to complete data operations.
- Script: A script is an extension of a batch processing file. It is a program that stores text. Generally, a computer script program is a combination of a series

of operations that control computers to perform operations. In the script program, certain logic branches can be implemented.

- Node: A node defines the operations performed on data.
- Resource: Resources refer to self-defined codes or text files that are uploaded by users and scheduled when node tasks are executed.
- Expression: Node parameter values in a node job can be dynamically generated based on the running environment by using Expression Language (EL). EL uses simple arithmetic and logic to calculate and reference embedded objects, including job objects and tool objects.
- Environment variable: An environment variable is an object with a specific name in the operating system. It contains information to be used by one or more applications.
- PatchData: PatchData refers to the instance that is generated in a period of time by a periodically scheduled job.

6.2 Data Management

6.2.1 Data Management Process

The data management function helps you quickly establish data models and provides you with data entities for script and job development. With data management, you can:

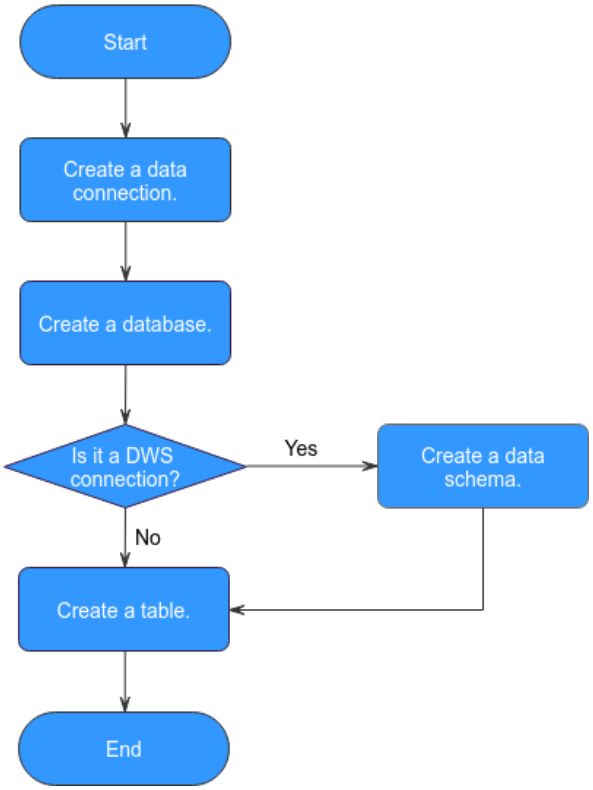
- Manage multiple types of data lakes, such as DWS, DLI, and MRS Hive.
- Use the GUI and DDL to manage database tables. If an MRS API connection is used, you cannot view or manage the databases, data tables, and fields of the connection in a visualized manner.
- Click  to view the databases, data tables, and fields in the data connection directory tree. The directory tree is only available for DWS SQL, DLI SQL, and Hive connections using an agent.

NOTE

If you have created a data connection, a database, and a data table, you can skip this section and go to **Script Development** or **Job Development**.

The following figure shows the process for using the data management function.

Figure 6-2 Data management process



1. Create a data connection to connect to a data lake base service. For details, see **Creating a Data Connection**.
2. Create a database based on the service type. For details, see **Creating a Database**.
3. If the connection type is DWS, create a database schema and a table. If the connection type is not DWS, create a table. For details, see **(Optional) Creating a Database Schema**.
4. Create a table. For details, see **Creating a Table**.

6.2.2 Creating a Data Connection

After a data connection is created, you can perform data operations on DataArts Factory, for example, managing databases, namespaces, database schema, and tables.

With one data connection, you can run multiple jobs and develop multiple scripts. If the connection information saved in the data connection changes, you only need to modify the corresponding information in Connection Management.

Creating a Data Connection

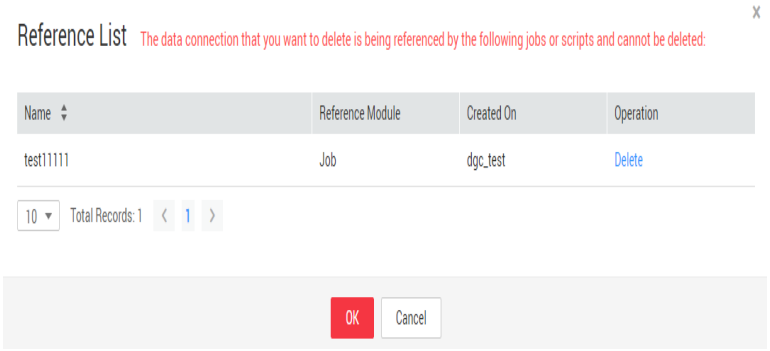
The data connection of DataArts Factory is created based on the data connection of Management Center. For details about how to create a data connection, see **Managing Data Connections**.

Viewing Connection References

To view the references of a connection, perform the following steps:

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. Click  to display the connection list.
5. Right-click a connection in the list and select **View Reference**.
6. In the displayed **Reference List** dialog box, view the references of the connection.

Figure 6-3 Reference List



6.2.3 Creating a Database

After creating a data connection, you can create a database on the console or using a SQL script.

- (Recommended) Console: You can directly create a database on the DataArts Studio DataArts Factory console with no code.
- SQL script: You can also develop and execute a SQL script for creating a database in the SQL editor of DataArts Studio's DataArts Factory module or a data lake product, and then use the script to create a database.

This section describes how to create a database on the DataArts Factory console.

Prerequisites

- You have already enabled the corresponding cloud services.
- A data connection has been created. For details, see **Creating a Data Connection**.
- MRS API connections cannot be used to manage databases in a visualized mode. You are advised to create a database using SQL scripts.
- Before deleting a database, ensure that the database is not in use and is not associated with any data tables.

Creating a Database on the DataArts Factory Console

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. In the script development menu, click  . Right-click the data connection for which you want to create a database, and choose **Create Database** from the shortcut menu. Set the parameters based on **Table 6-2**.

   Table 6-2 Creating a database

Parameter	Mandatory	Description
Database Name	Yes	Name of a database. The naming rules are as follows: • DLI: The value must contain only numbers, letters, and underscores (_). It cannot start with an underscore (_) or contain only numbers. • DWS: The value must contain only numbers, letters, and underscores (_). It cannot start with an underscore (_) or contain only numbers. • MRS Hive: The value must contain 1 to 128 characters, including only letters, numbers, and underscores (_). It must start with a number or letter and cannot contain only numbers.
Description	No	Descriptive information about the database. The requirements are as follows: • DLI: The value contains a maximum of 256 characters. • DWS: The value contains a maximum of 1,024 characters. • MRS Hive: The value contains a maximum of 1,024 characters.

5. Click **OK**.

Related Operations

- Modify a database: In the script development menu, click  . Expand a data connection, right-click a database name, select **Edit**, and modify the database information.
- Delete a database: In the script development menu, click  . Expand a data connection, right-click a database name, select **Delete**, and click **OK** in the displayed dialog box.

NOTE

Deleted databases cannot be recovered. Exercise caution when performing this operation.

6.2.4 (Optional) Creating a Database Schema

After creating a DWS data connection, you can manage the database schemas under the DWS data connection.

Prerequisites

- A DWS data connection has been created. For details, see **Creating a Data Connection**.
- A DWS database has been created. For details, see **Creating a Database**.

Creating a Database Schema

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. In the script development menu, click . Expand a DWS data connection, select the database to be configured, and expand the directory level to **schemas**. Then right-click **schemas** and select **Create Schema** from the shortcut menu.
5. In the displayed dialog box, set the schema parameters based on **Table 6-3**.

   Table 6-3 Creating a database schema

Parameter	Mandatory	Description
Mode Name	Yes	Name of a database schema.
Description	No	Descriptive information about the database schema.

6. Click **OK**.

Related Operations

- Modify a database schema: In the script development menu, click . Expand a data connection to the target database schema, right-click the database schema name, select **Edit**, and modify the database schema information.
- Delete a database schema: In the script development menu, click . Expand a data connection to the target database schema, right-click the database schema name, select **Delete**, and click **OK** in the displayed dialog box.

NOTE

- The default database schema cannot be deleted.
- Deleted database schemas cannot be recovered. Exercise caution when performing this operation.

6.2.5 Creating a Table

You can create a table on the DataArts Factory console, in DDL mode, or using a SQL script.

- (Recommended) Console: You can directly create a table on the DataArts Studio DataArts Factory console with no code.
- (Recommended) DDL mode: You can select the DDL mode in DataArts Studio's DataArts Factory mode to create a table using a SQL script.
- SQL script: You can also develop and execute a SQL script for creating a table in the SQL editor of DataArts Studio's DataArts Factory module or a data lake product, and then use the script to create a table.

This section describes how to create a table on the DataArts Factory console and in DDL mode.

Prerequisites

- You have created a database and a DWS database schema. For details, see **Creating a Database** and **(Optional) Creating a Database Schema**.
- A data connection that matches the table type has been created in DataArts Factory. For details, see **Creating a Data Connection**.

Creating a Table (GUI Mode)

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. In the script development menu, click , expand the data connection to **tables**, and right-click **Create Table** or click to create a table.
5. In the displayed dialog box, configure parameters based on **Table 6-4** on the **Basic property configuration** tab page.

Table 6-4 Basic property parameters

Data Connection Type	Description
DLI	For details, see the **Basic Property** part in **Table 6-8**.
DWS	For details, see the **Basic Property** part in **Table 6-9**.

Data Connection Type	Description
MRS Hive	For details, see the **Basic Property** part in **Table 6-10**.

6. Click **Next**. On the **Configure Table Structure** page, configure tbe table structure parameters based on **Table 6-5**.

Table 6-5 Table structure

Data Connection Type	Description
DLI	For details, see the **Table Structure** part in **Table 6-8**.
DWS	For details, see the **Table Structure** part in **Table 6-9**.
MRS Hive	For details, see the **Table Structure** part in **Table 6-10**.

7. Click **OK**.

Creating a Table (DDL Mode)

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. In the script development menu, click , expand the data connection to **tables**, and right-click **Create Table** or click to create a table.
5. Click **DDL-based Table Creation** and enter SQL statements in the displayed editor. (Default values are set for the parameters listed in **Table 6-6**.) The following is an example:

```
CREATE TABLE userinfo ( id INT, name STRING);
```

NOTE

The SQL syntax varies depending on the data source. Before developing an SQL statement, learn about the syntax of the data source from its documentation.

Table 6-6 Data table parameters

Parameter	Description
Data Connection Type	Type of data connection to which the table belongs
Data Connection	Data connection to which the table belongs

Parameter	Description
Database	Database where the data table is located

6. Click **Save**.

Related Operations

- View table details: In the script development menu, click . Expand the data connection to the data table level, right-click a table name, and select **View Details** from the shortcut menu to view the table details shown in **Table 6-7**.

  Table 6-7 Table details

Tab Name	Description
Table Information	Displays the basic information and storage information about the table.
Field Information	Displays the field information about the table.
Data Preview	Displays 10 records in the table.
DDL	Displays the DDL of the DWS, DLI, or MRS Hive data table.

- Delete a table: In the script development menu, click . Expand a data connection, right-click a table name, select **Delete**, and click **OK** in the displayed dialog box.

  NOTE

  Deleted tables cannot be recovered. Exercise caution when performing this operation.

Parameter Description

Table 6-8 DLI data table

Parameter	Mandatory	Description
Basic Property		
Table Name	Yes	Name of a table. The name must contain 1 to 63 characters, including only lowercase letters, numbers, and underscores (_). It cannot contain only numbers or start with an underscore.

Parameter	Mandatory	Description
Alias	No	Alias of a table. The alias must contain 1 to 63 characters, including only letters, numbers, and underscores (_). It cannot contain only numbers or start with an underscore.
Data Connection Type	Yes	Type of the data connection to which the table belongs. The default value is used and cannot be changed.
Data Connection	Yes	Data connection to which the table belongs. The default value is used and cannot be changed.
Database	Yes	Database where the data table is located. The default value is used and cannot be changed.
Data Location	Yes	Location to save data. Possible values: • OBS • DLI
Data Format	Yes	Format of data. This parameter is available only when **Data Location** is set to **OBS**. Possible values: • **parquet**: DataArts Factory can read non-compressed parquet data and parquet data compressed using Snappy or gzip. • **csv**: DataArts Factory can read non-compressed CSV data and CSV data compressed using gzip. • **orc**: DataArts Factory can read non-compressed ORC data and ORC data compressed using Snappy. • **json**: DataArts Factory can read non-compressed JSON data and JSON data compressed using gzip.

Parameter	Mandatory	Description
Path	Yes	OBS path where the data is stored. This parameter is available only when **Data Location** is set to **OBS**. If no OBS path or OBS bucket is available, the system automatically creates an OBS directory. **NOTE** If the number of OBS buckets has reached the upper limit, the system automatically displays the following message: "Failed to create the OBS directory. Error cause: [Create OBS Bucket failed:TooManyBuckets:You have attempted to create more buckets than allowed]".
Table Description	No	Descriptive information about the table.
Table Structure		
Column Type	Yes	Type of the column. Available options include **Partition Column** and **Common Column**. The default value is **Common Column**.
Column Name	Yes	Name of the column. The name must be unique.
Type	Yes	Type of data.
Column Description	No	Descriptive information about the column.
Operation	No	To add a column, click  . To delete a column, click  .

Table 6-9 DWS data table

Parameter	Mandatory	Description
Basic Property		
Table Name	Yes	Name of a table. The name must contain 1 to 63 characters, including only letters, numbers, and underscores (_). It cannot contain only numbers or start with an underscore.

Parameter	Mandatory	Description
Alias	No	Alias of a table. The alias must contain 1 to 63 characters, including only letters, numbers, and underscores (_). It cannot contain only numbers or start with an underscore.
Data Connection Type	Yes	Type of the data connection to which the table belongs. The default value is used and cannot be changed.
Data Connection	Yes	Data connection to which the table belongs. The default value is used and cannot be changed.
Database	Yes	Database where the data table is located. The default value is used and cannot be changed.
Schema	Yes	Schema of the database.
Table Description	No	Descriptive information about the table.
Advanced Settings	No	The following advanced options are available: ● Storage method of a table. Possible values: – **Row store** – **Column store** ● Compression level of a table – Available values when the storage method is row store: **YES** or **NO**. – Available values when the storage method is column store: **YES**, **NO**, **LOW**, **MIDDLE**, or **HIGH**. For the same compression level in column store mode, you can configure compression grades from 0 to 3. Within any compression level, the higher the grade, the greater the compression ratio.
Table Structure		
Column Name	Yes	Name of the column. The name must be unique.

Parameter	Mandatory	Description
Data Classification	Yes	Classification of data. Possible values: • **Value** • **Currency** • **Boolean** • **Binary** • **Character** • **Time** • **Geometric** • **Network address** • **Bit string** • **Text search** • **UUID** • **JSON** • **OID**
Data Type	Yes	Type of data.
Column Description	No	Descriptive information about the column.
Create ES Index	No	If you click the check box, an ES index needs to be created. When creating the ES index, select the created CSS cluster from the **CloudSearch Cluster Name** drop-down list. For details about how to create a CSS cluster, see *Cloud Search Service User Guide*.
Index Data Type	No	Data type of the ES index. The options are as follows: • text • keyword • date • long • integer • short • byte • double • boolean • binary
Operation	No	To add a column, click  . To delete a column, click  .

Table 6-10 MRS Hive data table

Parameter	Mandatory	Description
Basic Property		
Table Name	Yes	Name of a table. The name must contain 1 to 63 characters, including only lowercase letters, numbers, and underscores (_). It cannot contain only numbers or start with an underscore.
Alias	No	Alias of a table. The alias must contain 1 to 63 characters, including only letters, numbers, and underscores (_). It cannot contain only numbers or start with an underscore.
Data Connection Type	Yes	Type of the data connection to which the table belongs. The default value is used and cannot be changed.
Data Connection	Yes	Data connection to which the table belongs. The default value is used and cannot be changed.
Database	Yes	Database to which the table belongs. The default value is used and cannot be changed.
Table Description	No	Descriptive information about the table.
Table Structure		
Column Name	Yes	Name of the column. The name must be unique.
Data Classification	Yes	Classification of data. Possible values: • Original type • ARRAY • MAP • STRUCT • UNION
Data Type	Yes	Type of data. See **LanguageManual DDL**.
Column Description	No	Descriptive information about the column.
Operation	No	To add a column, click  . To delete a column, click  .

6.3 Script Development

6.3.1 Script Development Process

The script development function provides the following capabilities:

- Provides an online script editor for developing and debugging SQL, Python, and Shell scripts.
- Supports script import and export.
- Allows use of variables and functions.
- Provides editing locks for collaborative development.
- Supports script version management and generation of saved and submitted versions.

  NOTE

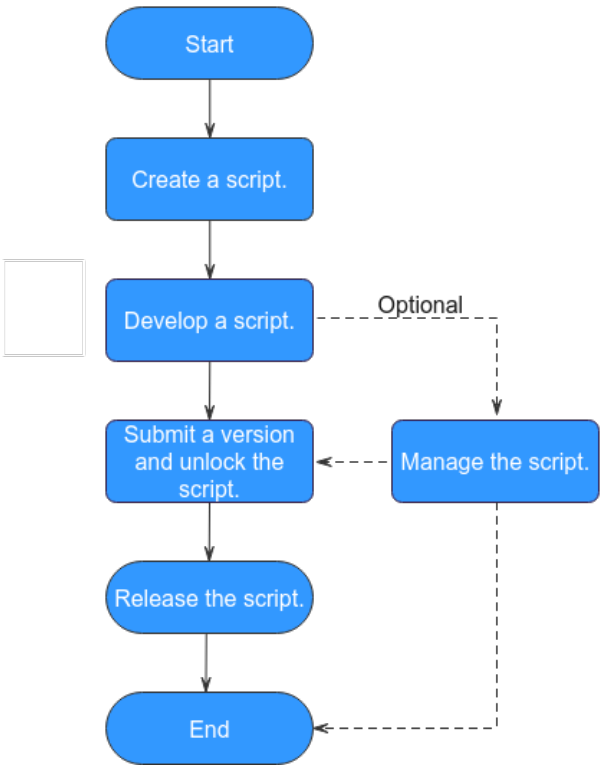
  If you save a script multiple times within a minute, only one version is recorded. If the intermediate data is important, you can click **Save new version** to save and add a version.

- Allows you to copy long script names. Click  , perform fuzzy search to query matched scripts, and click the copy button next to a long script name to copy it.
- Allows you to right-click a script to quickly copy the script name and to quickly close an opened script tab page.
- Provides a link in the execution results of MRS Spark SQL and MRS Hive SQL scripts that use a connection of the MRS API type. Through this link, you can switch to MRS Yarn to view execution logs.
- Allows you to switch to the task release page by placing the cursor on  and clicking **Release** when developing a script in enterprise mode.
- Allows you to filter submitted and unsubmitted scripts. Unsubmitted scripts are marked in red.
- Displays script parameters in a dialog box. Parameter values can be changed, but parameter names cannot. You can click **Test Parameters** to view (but not modify) the parameters referenced by the script and the environment variables configured in the environment. Parameters in the SQL statement can be sorted by name.
- Supports configuration of the SQL editor style. Move the cursor to  and click **Style Configuration** to configure the editor, icon display, annotation templates, and shortcut keys that can be used in the SQL script editor.
- Allows you to view SQL query results in a table or list. You can click **Style Configuration** and set **SQL Query Result Display Mode** on the **Configure Editor** tab page.
- Allows you to go to the release page from the script development page in enterprise mode. Place the cursor over  and click **Release** to go to the task release page.

- Allows you to view tables of Hive SQL, DLI SQL, RDS SQL, and DWS SQL scripts. You can expand a table name to view the column names, field types, and descriptions in the table.

The following figure shows the process of script development.

Figure 6-4 Script development process



1. Create a script of the corresponding type. For details, see **Creating a Script**.
2. Develop the script: Develop, debug, and execute the script online. For details, see **Developing Scripts**.
3. Submit a version and unlock the script: After performing this step, the script can be scheduled by jobs and modified by other developers. For details, see **Submitting a Version**.
4. (Optional) Manage the script: After the script development is complete, you can manage the script as required. For details, see **(Optional) Managing Scripts**.
5. Release the script. This step is required in enterprise mode. For details, see **Releasing a Script Task**.

6.3.2 Creating a Script

DataArts Factory allows you to create, edit, debug, and run SQL, Python, and Shell scripts. Before developing a script, you must create one.

Prerequisites

You have completed operations in **Creating a Data Connection** and **Creating a Database**.

Procedure

Creating a Directory (If a directory already exists, you do not need to create one.)

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. In the script directory list, right-click a directory and choose **Create Directory** from the shortcut menu.
5. In the displayed dialog box, configure directory parameters. **Table 6-11** describes the directory parameters.

   Table 6-11 Script directory parameters

Parameter	Description
Directory Name	Name of the script directory. The name must contain 1 to 64 characters, including only letters, numbers, underscores (_), and hyphens (-).
Select Directory	Parent directory of the script directory. The parent directory is the root directory by default.

6. Click **OK**.

Creating a Script

1. In the script directory list, right-click a directory and select **Create** *Script type* **Script** from the shortcut menu.
2. Go to the script development page. For details, see **Developing an SQL Script**, **Developing a Shell Script**, and **Developing a Python Script**.

   NOTE

   A maximum of five temporary scripts of the same type can be created. If you close a temporary script without saving it and create a script of the same type, the closed temporary script will be opened again.

6.3.3 Developing Scripts

6.3.3.1 Developing an SQL Script

DataArts Factory allows you to develop, debug, and run SQL scripts online. You can run developed scripts in jobs. For details, see **Developing a Pipeline Job**.

DataArts Factory supports the following types of SQL scripts. The SQL syntax varies depending on the data source. Before developing an SQL statement, learn about the syntax of the corresponding data source.

- DLI SQL scripts: For details, see **Spark SQL Syntax**.
- Hive SQL scripts: For details, see **Hive SQL**.
- DWS SQL scripts: For details, see **About GaussDB(DWS) SQL**.
- Spark SQL scripts: For details, see **Spark2x Basic Principles**.
- Flink SQL scripts: For details, see **Stream SQL Join**.
- RDS SQL scripts: For details, see **Syntax**.
- Presto SQL scripts: For details, see **Presto**.
- Spark Python scripts: For details, see **Spark2x Basic Principles**.

Prerequisites

- A corresponding cloud service has been enabled and a database has been created in the cloud service.
- A data connection that matches the data connection type of the created script. For details, see **Managing Data Connections**. The Flink SQL script does not involve this operation.
- You have locked the script. Otherwise, you must click **Lock** so that you can develop the script. A script you create or import is locked by you by default. For details, see the **lock function**.

Procedure

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. In the script directory, double-click a script to access the script development page.
5. In the upper part of the editor, select script properties. **Table 6-12** describes the script properties. Skip this step when creating a Flink SQL script.

Table 6-12 SQL script properties

Property	Description
Data Connection	Select a data connection.
Database	Name of the database.

Property	Description
Resource Queue	Selects a resource queue for executing a DLI job. Set this parameter when a DLI or SQL script is created. You can create a resource queue using either of the following methods: ● Click ⊕. The **Buy Queue** page of DLI is displayed. ● Go to the DLI console. **NOTE** The default resource queue **default** provided by DLI is only used for trial. It may be occupied by multiple users at a time. Therefore, it is possible that you fail to obtain the resource for related operations. If the execution takes a long time or fails, you are advised to try again during off-peak hours or use a self-built queue to run the job. In addition, the **default** queue does not support the insert, load, or cat commands. To set properties for submitting SQL jobs in the form of **key/value**, click ⚙. A maximum of 10 properties can be set. The properties are described as follows: **NOTE** ● The environment variable must start with **dli.sql.** or **spark.sql.** ● If the key of the environment variable is **dli.sql.shuffle.partitions** or **dli.sql.autoBroadcastJoinThreshold**, the environment variable cannot contain the greater than (>) or less than (<) sign. ● If a parameter with the same name is configured in both a job and a script, the parameter value configured in the job will overwrite that configured in the script. ● **dli.sql.autoBroadcastJoinThreshold**: specifies the data volume threshold to use BroadcastJoin. If the data volume exceeds the threshold, BroadcastJoin will be automatically enabled. ● **dli.sql.shuffle.partitions**: specifies the number of partitions during shuffling. ● **dli.sql.cbo.enabled**: specifies whether to enable the CBO optimization policy. ● **dli.sql.cbo.joinReorder.enabled**: specifies whether join reordering is allowed when CBO optimization is enabled. ● **dli.sql.multiLevelDir.enabled**: specifies whether to query the content in subdirectories if there are subdirectories in the specified directory of an OBS table or in the partition directory of an OBS partition table. By default, the content in subdirectories is not queried. ● **dli.sql.dynamicPartitionOverwrite.enabled**: specifies that only partitions used during data query are overwritten and other partitions are not deleted.

Property	Description
	NOTE When you run a DLI SQL script or test a DLI SQL single-task job in non-scheduling scenarios, the following parameters are enabled by default: • **spark.sql.adaptive.enabled**: Adaptive Query Execution (AQE) is enabled so that Spark can dynamically optimize the query execution plan based on the characteristics of the data being processed and improve the performance by reducing the amount of data to be processed. • **spark.sql.adaptive.join.enabled**: AQE is enabled for join operations. The optimal join algorithm is selected based on the data being processed to improve performance. • **spark.sql.adaptive.skewedJoin.enabled**: AQE is enabled for skewed join operations. Skewed data can be automatically detected and the join algorithm is optimized accordingly to improve performance. • **spark.sql.mergeSmallFiles.enabled**: Merging of small files is enabled. Small files can be merged into large ones, improving performance and shortening the processing time. In addition, less files need to be read from remote storage, and more local files can be used. If you do not want to use these functions, you can set the values of the preceding parameters to **false**.

6. Enter an SQL statement in the editor. You can enter multiple SQL statements.

   The SQL syntax varies depending on the data source. Before developing an SQL statement, learn about the syntax of the corresponding data source.

   - DLI SQL scripts: For details, see **Spark SQL Syntax**.
   - Hive SQL scripts: For details, see **Hive SQL**.
   - DWS SQL scripts: For details, see **About GaussDB(DWS) SQL**.
   - Spark SQL scripts: For details, see **Spark2x Basic Principles**.
   - Flink SQL scripts: For details, see **Stream SQL Join**.
   - RDS SQL scripts: For details, see **Syntax**.
   - Presto SQL scripts: For details, see **Presto**.
   - Spark Python scripts: For details, see **Spark2x Basic Principles**.

NOTE

- SQL statements are separated by semicolons (;). If semicolons are used in other places but not used to separate SQL statements, escape them with backslashes (\). For example:
```
select 1;
select * from a where b="dsfa\;";  --example 1\;example 2.
```
- RDS SQL does not support the begin ... commit transaction syntax. If necessary, use the start transaction ... commit transaction syntax.
- The script cannot be larger than 16 MB.
- The system date obtained by using an SQL statement is different from that obtained by using the database tool. The query result is stored in the database in the YYYY-MM-DD format, but the query result displayed on the page is in the converted format.
- When a user submits a Spark SQL script to MRS, the script is submitted to the tenant queue bound to the user by default. The bound queue is the queue corresponding to tenant role of the user. If there are multiple queues, the system preferentially selects a queue based on the queue priorities. To set a fixed queue for the user to submit scripts, log in to FusionInsight Manager, choose **Tenant Resources** > **Dynamic Resource Plan**, and click the **Global User Policy** tab. For details, see **Managing Global User Policies**.

To facilitate script development, DataArts Factory provides the following capabilities:

– The script editor supports the following shortcut keys, which improve the script development efficiency:
  - **F8**: Run a script.
  - **F9**: Stop running a script.
  - **Ctrl** + **/**: Comment out or uncomment the line or code block where the cursor resides.
  - **Ctrl** + **S**: Save a script.
  - **Ctrl** + **Z**: Undo an action.
  - **Ctrl** + **F**: Search for information.
  - **Ctrl** + **Shift** + **R**: Replace
  - **Ctrl** + **X**: Cut (Cut a line when the cursor selects nothing.)
  - **Alt** + mouse dragging: Select columns to edit a block.
  - **Ctrl** + mouse click: Select multiple lines to edit or indent them together.
  - **Shift** + **Ctrl** + **K**: Delete the current line.
  - **Ctrl** + → (or ←): Move the cursor rightwards (or leftwards) by word.
  - **Ctrl** + **Home** or **Ctrl** + **End**: Navigate to the beginning or end of the current file.
  - **Home** or **End**: Navigate to the beginning or end of the current line.

- **Ctrl** + **Shift** + **L**: Double-click all the same character strings and add cursors to them to implement batch modification.
- **Ctrl** + **D**: Delete a line.
- **Shift** + **Ctrl** + **U**: Unlock a script.
- **Ctrl** + **Alt** + **K**: Select the word where the cursor resides.
- **Ctrl** + **B**: Format
- **Ctrl** + **Shift** + **Z**: Redo
- **Ctrl** + **Enter**: Execute the selected line or content.
- **Ctrl** + **Alt** + **F**: Flag
- **Ctrl** + **Shift** + **K**: Search for the previous one.
- **Ctrl** + **K**: Search for the next one.
- **Ctrl** + **Backspace**: Delete the word to the left of the cursor.
- **Ctrl** + **Delete**: Delete the word to the right of the cursor.
- **Alt** + **Backspace**: Delete all content from the beginning of the line to the cursor.
- **Alt** + **Delete**: Delete all content from the cursor to the end of the line.
- **Alt** + **Shift**-**Left**: Select all content from the beginning of the line to the cursor.
- **Alt** + **Shift**-**Right**: Select all content from the cursor to the end of the line.

– System functions (Flink SQL, Spark SQL, ClickHouse SQL, and Presto SQL do not support system functions.)

  To view the functions supported by this type of data connection, click **System Functions** on the right of the editor. You can double-click a function to the editor to use it.

– Data tables can be read to generate SQL statements. (Flink SQL, Spark SQL, ClickHouse SQL, and Presto SQL do not support this function.)

  Click **Data Tables** on the right of the editor to display all the tables in the current database or schema. You can select tables and columns and click **Generate SQL Statement** in the lower right corner to generate an SQL statement, which you need to manually format.

– Script parameters (Currently, only Flink SQL does not support script parameters.)

  You can directly write script parameters in SQL statements. When debugging scripts, you can enter parameter values in the script editor. If the script is referenced by a job, you can set parameter values on the job development page. The parameter values can use EL expressions (see **Expression Overview**).

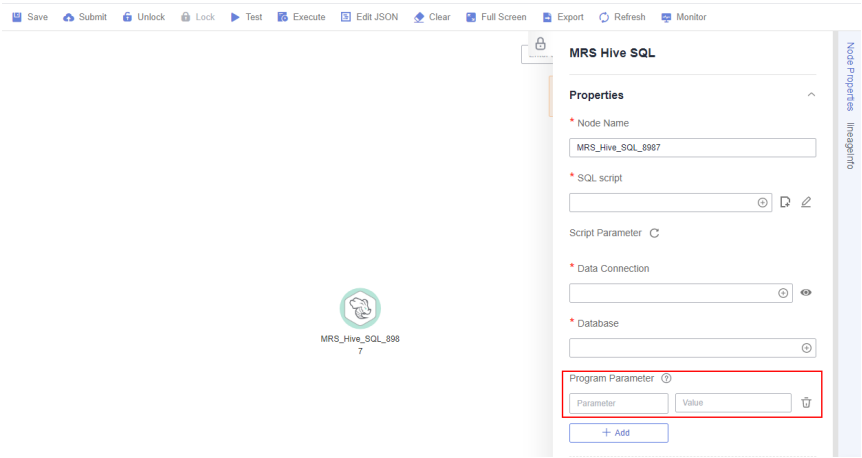
> **NOTE**
>
> If a parameter in an SQL script involves a variable, the format of the variable must be the same as that set in **Configuring Script Variables**. If they are different, the variable cannot be identified.

In the following script example, *str1* indicates the parameter name. It can contain only letters, digits, hyphens (-), underscores (_), greater-than signs (>), and less-than signs (<), and can contain a maximum of 16 characters. The parameter name must be unique.

```
select ${str1} from data;
```

For MRS Spark SQL and MRS Hive SQL scripts, you set a program parameter by referring to **set hive.exec.parallel=true;** in the SQL statements or configure this parameter by setting **Program Parameter** on **Node Properties** of the job.

Figure 6-5 Program Parameter



- Owner

  Click **Basic Info** to set the script owner and description.

- Allows you to go to the release page from the script development page in enterprise mode. Place the cursor over ☰ and click **Release**.
- For MRS API connections, parameters and values can be configured for Spark SQL and Hive SQL scripts. For proxy connections, this function is not supported.

NOTE

Click in the upper right corner to set environment variables for scripts. The following are some examples:

Set environment variables for a Hive SQL script:

--hiveconf hive.merge.mapfiles=true;

--hiveconf mapred.job.queue.name=queue1

Set environment variables for a Spark SQL script:

--num-executors 1

--executor-cores 4

--queue queue2

The former indicates the parameter name, and the latter indicates the parameter value.

After the script is executed, view the execution details on the MRS management plane.

7. (Optional) In the upper part of the editor, click **Format** to format SQL statements. When developing a Flink SQL script, skip this step.
8. In the upper part of the editor, click **Execute**. If you need to execute some SQL statements separately, select the SQL statements first. After executing the SQL statements, view the execution history and result of the script in the lower part of the editor. When developing a Flink SQL script, skip this step.

   NOTE

   - A maximum of 1,000 SQL statement execution results can be displayed. A maximum of 10,000 DLI SQL statement execution results can be displayed. To view more execution results, download or dump them by following the instructions in **Downloading or Dumping a Script Execution Result**.
   - You can perform the following operations on execution results:
     - Double-click or right-click the name of an execution result tab to rename it. The name can contain a maximum of 16 characters.
     - Right-click the name of an execution result tab to close the current tab, all the tabs to the left or right of the current tab, all the other tabs, or all the tabs.
   - If the MRS cluster is a non-security cluster and the command whitelist is not restricted, you can easily find the corresponding task on the Yarn management page of MRS based on the script name and execution time after adding the application name information during Hive SQL execution. Note that if the default engine is **tez**, you need to set the engine to **mr** to disable the tez engine.
9. Above the editor, click to save the script.

   If the script is created but not saved, set the parameters listed in **Table 6-13**.

Table 6-13 Script parameters

Parameter	Mandatory	Description
Script Name	Yes	Name of the script. The name contains a maximum of 128 characters, including only letters, numbers, hyphens (-), underscores (_), and periods (.).

Parameter	Mandatory	Description
Owners	No	Owner of the script. By default, the creator of the script is the owner.
Description	No	Descriptive information about the script.
Select Directory	Yes	Directory to which the script belongs. The root directory is selected by default.

NOTE

If you open an unsaved script, you can restore its content from the local cache.

After the script is saved, a version is automatically generated and displayed in **Versions**. The version can be rolled back. If you save a script multiple times within a minute, only one version is recorded. If the intermediate data is important, you can click **Save new version** to save and add a version.

Downloading or Dumping a Script Execution Result

After a script is executed successfully, you can download or dump the execution result. By default, all users can download and dump the execution results of SQL scripts. If you do not want all users to have this permission, configure the permission by referring to **Configuring a Data Export Policy**.

- After executing a script, you can click **Download** on the **Result** tab page to download a CSV result file to a local path. You can view the download record on the **Download Center** page.
- After executing a script, you can click **Dump** on the **Result** tab page to dump a CSV and a JSON result file to OBS. For details, see **Table 6-14**.

  NOTE

  - The dump function is supported only if the OBS service is available.
  - Only the execution results of the query statements in SQL scripts can be dumped.

Table 6-14 Dump parameters

Parameter	Mandatory	Description
Data Format	Yes	Format of the data to be exported. CSV and JSON formats are supported.
Resource Queue	No	DLI queue where the export operation is to be performed. Set this parameter when a DLI or SQL script is created.

Parameter	Mandatory	Description
Compression Format	No	Format of compression. Set this parameter when a DLI or SQL script is created. • none • bzip2 • deflate • gzip
Storage Path	Yes	OBS path where the result file is stored. After selecting an OBS path, customize a folder. Then, the system will create it automatically for storing the result file. You can also go to the **Download Center** page to set the default OBS path, which will be automatically set for **Storage Path** in the **Dump Result** dialog box.
Cover Type	No	If a folder that has the same name as your custom folder exists in the storage path, select a cover type. Set this parameter when a DLI or SQL script is created. • **Overwrite**: The existing folder will be overwritten by the customized folder. • **Report**: The system reports an error and suspends the export operation.
Export Column Name	No	**Yes**: Column names will be exported. **No**: Column names will not be exported.
Character Set	No	• **UTF-8**: default character set • **GB2312**: recommended when the data to be exported contains Chinese character sets • **GBK**: expanded based on and compatible with GB2312

Download or dump allows you to view more SQL script execution results. **Table 6-15** lists the maximum number of results that can be viewed, dumped, and downloaded for different types of SQL scripts.

Table 6-15 Maximum number of results that can be viewed, dumped, and downloaded

SQL Type	Maximum Number of Results That Can Be Viewed Online	Maximum Number of Results That Can Be Downloaded	Maximum Number of Results That Can Be Dumped
DLI	10000	1000	Unlimited
Hive	1000	1000	10000
DWS	1000	1000	10000
Spark	1000	1000	10000
RDS	1000	1000	Not supported
Presto	1000	The downloaded results are directly dumped to OBS. The number of results is unlimited.	Unlimited
ClickHouse	1000	1000	10000
HetuEngine	1000	1000	10000
Impala	1000	1000	10000

6.3.3.2 Developing a Shell Script

DataArts Factory allows you to develop, debug, and run Shell scripts online. You can run developed scripts in jobs. For details, see **Developing a Pipeline Job**.

Prerequisites

- A shell script has been added. For details, see **Creating a Script**.
- A host connection has been created. The Linux host is used to execute shell scripts. For details, see **Configuring a Host Connection**.
- You have the permission to create and execute files in the **/tmp** directory on the host.
- The maximum number of shell or Python scripts that can run concurrently on the ECS is determined by the value of **MaxSessions** in the **/etc/ssh/sshd_config** file on the ECS. Set **MaxSessions** based on the scheduling frequency of shell or Python scripts.
- You have locked the script. Otherwise, you must click **Lock** so that you can develop the script. A script you create or import is locked by you by default. For details, see the **lock function**.

Procedure

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. In the script directory list, double-click a script that you want to develop. The script development page is displayed.
5. In the upper part of the editor, select script properties. **Table 6-16** describes the script properties.

Table 6-16 Shell script properties

Parameter	Description
Host Connection	Selects the host where a shell script is to be executed.

Click **Input Parameters** and enter the parameter and interactive parameter for executing the shell script.

Table 6-17 Shell script parameters

Parameter	Description
Parameter	Parameter transferred to the Shell script when it is executed. Parameters are separated by spaces, for example, **a b c**. The parameter must be referenced by a location variable (for example, $1, $2, or $3) in the Shell script. Otherwise, the parameter is invalid. The location variable starts from 0. Variable 0 is reserved for storing the actual script name, variable 1 corresponds to the first parameter of the script, and so on. For example, $1, $2, and $3 reference parameters **a**, **b**, and **c**, respectively. Note: If a variable is referenced in the shell script, use the *$args* format instead of the *${args}* format. Otherwise, the variable will be replaced by a parameter with the same name in the job. For example, if you enter **a b c** and run the following Shell script, **b** is displayed: `echo $2`

Parameter	Description

Interactive Parameter	Interactive information (for example, passwords) provided during shell script execution. Interactive parameters are separated by spaces. The shell script reads parameter values in sequence according to the interaction situation.
	For example, run the following interactive Shell script. Interaction parameters **1**, **2**, and **3** correspond to **begin**, **end**, and **exit**, respectively.
	● When the interaction parameter is set to **1**, the execution result is **start something**.
	● When the interaction parameter is set to **2**, the execution result is **stop something**.
	● When the interaction parameter is set to **3**, the execution result is **exit**.
	`#!/bin/bash`
	`select Actions in "begin" "end" "exit"`
	`do`
	`case $Actions in`
	`"begin")`
	`echo "start something"`
	`break`
	`;;`
	`"end")`
	`echo "stop something"`
	`break`
	`;;`
	`"exit")`
	`echo "exit"`
	`break`
	`;;`
	`*)`
	`echo "Ignorant"`
	`;;`
	`esac`
	`done`
	The following is an example of using the read -p syntax:
	`read -p "Parameter 1 and parameter 2"Variable 1 Variable 2`

6. Edit shell statements in the editor. To facilitate script development, DataArts Factory provides the following capabilities:
   - The script editor supports the following shortcut keys, which improve the script development efficiency:
     - **F8**: Run a script.
     - **F9**: Stop running a script.
     - **Ctrl** + **/**: Comment out or uncomment the line or code block where the cursor resides.
     - **Ctrl** + **S**: Save a script.
     - **Ctrl** + **Z**: Undo an action.
     - **Ctrl** + **F**: Search for information.

- **Ctrl** + **Shift** + **R**: Replace
- **Ctrl** + **X**: Cut (Cut a line when the cursor selects nothing.)
- **Alt** + mouse dragging: Select columns to edit a block.
- **Ctrl** + mouse click: Select multiple lines to edit or indent them together.
- **Shift** + **Ctrl** + **K**: Delete the current line.
- **Ctrl** + → (or ←): Move the cursor rightwards (or leftwards) by word.
- **Ctrl** + **Home** or **Ctrl** + **End**: Navigate to the beginning or end of the current file.
- **Home** or **End**: Navigate to the beginning or end of the current line.
- **Ctrl** + **Shift** + **L**: Double-click all the same character strings and add cursors to them to implement batch modification.
- **Ctrl** + **D**: Delete a line.
- **Shift** + **Ctrl** + **U**: Unlock a script.
- **Ctrl** + **Alt** + **K**: Select the word where the cursor resides.
- **Ctrl** + **B**: Format
- **Ctrl** + **Shift** + **Z**: Redo
- **Ctrl** + **Enter**: Execute the selected line or content.
- **Ctrl** + **Alt** + **F**: Flag
- **Ctrl** + **Shift** + **K**: Search for the previous one.
- **Ctrl** + **K**: Search for the next one.
- **Ctrl** + **Backspace**: Delete the word to the left of the cursor.
- **Ctrl** + **Delete**: Delete the word to the right of the cursor.
- **Alt** + **Backspace**: Delete all content from the beginning of the line to the cursor.
- **Alt** + **Delete**: Delete all content from the cursor to the end of the line.
- **Alt** + **Shift**-**Left**: Select all content from the beginning of the line to the cursor.
- **Alt** + **Shift**-**Right**: Select all content from the cursor to the end of the line.

– Script parameter function. Use this function in either of the following ways:

i. Write the script parameter name and parameter value in the shell statement. When the shell script is referenced by a job, if the parameter name configured for the job is the same as the parameter name of the shell script, the parameter value of the shell script is replaced by the parameter value of the job.

An example is as follows:

```
a=1
echo ${a}
```

In the preceding command, *a* indicates the parameter name. It can contain only letters, digits, hyphens (-), underscores (_), greater-than signs (>), and less-than signs (<), and can contain a maximum of 16 characters. The parameter name must be unique.

ii. Configure parameters in the upper part of the editor. When you execute the shell script, the configured parameters are transferred to the script. Separate parameters by spaces, for example, **a b c**. The parameter must be referenced by the shell script. Otherwise, the parameter is invalid.

Note: If a variable is referenced in the shell script, use the *$args* format instead of the *${args}* format. Otherwise, the variable will be replaced by a parameter with the same name in the job.

- Owner

  Click **Basic Info** to set the script owner and description.
- The script cannot be larger than 16 MB.
- Allows you to go to the release page from the script development page in enterprise mode. Place the cursor over ☰ and click **Release**.

7. In the lower part of the editor, click **Execute**. After executing the shell statement, view the execution history and result of the script in the lower part of the editor.

   NOTE

   You can perform the following operations on execution results:

   - Double-click or right-click the name of an execution result tab to rename it. The name can contain a maximum of 16 characters.
   - Right-click the name of an execution result tab to close the current tab, all the tabs to the left or right of the current tab, all the other tabs, or all the tabs.
   - The execution result of a Shell script cannot be larger than 30 MB. Otherwise, an error is reported.

8. Above the editor, click 💾 to save the script.

   If the script is created but not saved, set the parameters listed in **Table 6-18**.

Table 6-18 Script parameters

Parameter	Mandatory	Description
Script Name	Yes	Name of the script. The name contains a maximum of 128 characters, including only letters, numbers, hyphens (-), underscores (_), and periods (.).

Parameter	Mandatory	Description
Description	No	Description of the script
Select Directory	Yes	Directory to which the script belongs. The root directory is selected by default.

NOTE

If you open an unsaved script, you can restore its content from the local cache.

After the script is saved, a version is automatically generated and displayed in **Versions**. The version can be rolled back. If you save a script multiple times within a minute, only one version is recorded. If the intermediate data is important, you can click **Save new version** to save and add a version.

6.3.3.3 Developing a Python Script

DataArts Factory allows you to develop, debug, and run Python scripts online. You can run developed scripts in jobs. For details, see **Developing a Pipeline Job**.

For details about how to develop a Python scripts, see **Developing a Python Script**.

Prerequisites

- A Python script has been added. For details, see **Creating a Script**.
- A host connection has been created. The Linux host is used to execute Python scripts. For details about how to create a host connection, see **Configuring a Host Connection**.
- You have the permission to create and execute files in the **/tmp** directory on the host.
- The maximum number of shell or Python scripts that can run concurrently on the ECS is determined by the value of **MaxSessions** in the **/etc/ssh/sshd_config** file on the ECS. Set **MaxSessions** based on the scheduling frequency of shell or Python scripts.
- You have locked the script. Otherwise, you must click **Lock** so that you can develop the script. A script you create or import is locked by you by default. For details, see the **lock function**.

Procedure

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. In the script directory list, double-click a script that you want to develop. The script development page is displayed.

5. In the upper part of the editor, configure the Python version and the host connection for executing the Python script.

   Table 6-19 Python script properties

Parameter	Description
Python Version	Select a Python version. • **Python2**: indicates Python 2. • **Python3**: indicates Python 3.
Host Connection	Select the host where a Python script is to be executed.

   Click **Input Parameters** and enter the parameter and interactive parameter for executing the Python script.

   Table 6-20 Python script parameters

Parameter	Description
Parameter	Parameter transferred to the Python script when the script is executed. Parameters are separated by spaces, for example, **a b c**. The parameter must be referenced by the Python script. Otherwise, the parameter is invalid.
Interactive Parameter	Interactive information (for example, passwords) provided during Python script execution. Interactive parameters are separated by spaces. The Python statement reads parameter values in sequence according to the interaction situation.

6. Edit Python statements in the editor. To facilitate script development, DataArts Factory provides the following capabilities:
   - The script editor supports the following shortcut keys, which improve the script development efficiency:
     - **F8**: Run a script.
     - **F9**: Stop running a script.
     - **Ctrl** + **/**: Comment out or uncomment the line or code block where the cursor resides.
     - **Ctrl** + **S**: Save a script.
     - **Ctrl** + **Z**: Undo an action.
     - **Ctrl** + **F**: Search for information.
     - **Ctrl** + **Shift** + **R**: Replace

- **Ctrl** + **X**: Cut (Cut a line when the cursor selects nothing.)
- **Alt** + mouse dragging: Select columns to edit a block.
- **Ctrl** + mouse click: Select multiple lines to edit or indent them together.
- **Shift** + **Ctrl** + **K**: Delete the current line.
- **Ctrl** + → (or ←): Move the cursor rightwards (or leftwards) by word.
- **Ctrl** + **Home** or **Ctrl** + **End**: Navigate to the beginning or end of the current file.
- **Home** or **End**: Navigate to the beginning or end of the current line.
- **Ctrl** + **Shift** + **L**: Double-click all the same character strings and add cursors to them to implement batch modification.
- **Ctrl** + **D**: Delete a line.
- **Shift** + **Ctrl** + **U**: Unlock a script.
- **Ctrl** + **Alt** + **K**: Select the word where the cursor resides.
- **Ctrl** + **B**: Format
- **Ctrl** + **Shift** + **Z**: Redo
- **Ctrl** + **Enter**: Execute the selected line or content.
- **Ctrl** + **Alt** + **F**: Flag
- **Ctrl** + **Shift** + **K**: Search for the previous one.
- **Ctrl** + **K**: Search for the next one.
- **Ctrl** + **Backspace**: Delete the word to the left of the cursor.
- **Ctrl** + **Delete**: Delete the word to the right of the cursor.
- **Alt** + **Backspace**: Delete all content from the beginning of the line to the cursor.
- **Alt** + **Delete**: Delete all content from the cursor to the end of the line.
- **Alt** + **Shift**-**Left**: Select all content from the beginning of the line to the cursor.
- **Alt** + **Shift**-**Right**: Select all content from the cursor to the end of the line.

– Script parameter function. Use this function in either of the following ways:

  i. Write the script parameter name and parameter value in the Python statement. When the Python script is referenced by a job, if the

parameter name configured for the job is the same as the parameter name of the Python script, the parameter value of the Python script is replaced by the parameter value of the job.

The following is an example script:

```
a=1
print {a}
```

In the preceding command, *a* indicates the parameter name. It can contain only letters, digits, hyphens (-), underscores (_), greater-than signs (>), and less-than signs (<), and can contain a maximum of 16 characters. The parameter name must be unique.

ii. Click **Input Parameters** and set parameters, which will be transferred to the Python script during execution of the script. Separate parameters by spaces, for example, **a b c**. The parameter must be referenced by the Python script. Otherwise, the parameter is invalid.

- Owner

  Click **Basic Info** to set the script owner and description.

- The script cannot be larger than 16 MB.
- Allows you to go to the release page from the script development page in enterprise mode. Place the cursor over ☰ and click **Release**.

7. In the upper part of the editor, click **Execute**. After executing the Python statement, view the execution history and result of the script in the lower part of the editor.

   NOTE

   You can perform the following operations on execution results:

   - Double-click or right-click the name of an execution result tab to rename it. The name can contain a maximum of 16 characters.
   - Right-click the name of an execution result tab to close the current tab, all the tabs to the left or right of the current tab, all the other tabs, or all the tabs.
   - The execution result of a Python script cannot be larger than 30 MB. Otherwise, an error is reported.

8. Above the editor, click 💾 to save the script.

   If the script is created but not saved, set the parameters listed in **Table 6-21**.

Table 6-21 Script parameters

Parameter	Mandatory	Description
Script Name	Yes	Name of the script. The name contains a maximum of 128 characters, including only letters, numbers, hyphens (-), underscores (_), and periods (.).
Description	No	Description of the script
Select Directory	Yes	Directory to which the script belongs. The root directory is selected by default.

NOTE

If you open an unsaved script, you can restore its content from the local cache.

After the script is saved, a version is automatically generated and displayed in **Versions**. The version can be rolled back. If you save a script multiple times within a minute, only one version is recorded. If the intermediate data is important, you can click **Save new version** to save and add a version.

6.3.4 Submitting a Version

Submitting a version depends on the version management function of DataArts Factory.

Version management traces script and job changes, and supports version comparison and rollback. The system retains 100 latest version records. In addition, version management can be used to distinguish the development state and production state.

- Development state: Scripts or jobs have not been submitted and are used for debugging. In the development state, you can edit, save, and run scripts or jobs without affecting those being scheduled. In addition, when a job is being associated with a script or job dependency is being configured, the associated script or job will read the configuration in the development state.
- Production state: Script or jobs have been submitted and are used for formal scheduling. In formal scheduling, the latest submitted versions of scripts or jobs will be used in scenarios such as script invocation, instance rerunning, and job dependency and patch data configuration.

Prerequisites

A script has been developed.

Submitting a Script Version

If you submit a version, the latest script in the development state will be saved and submitted and overwrite the previous script version.

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

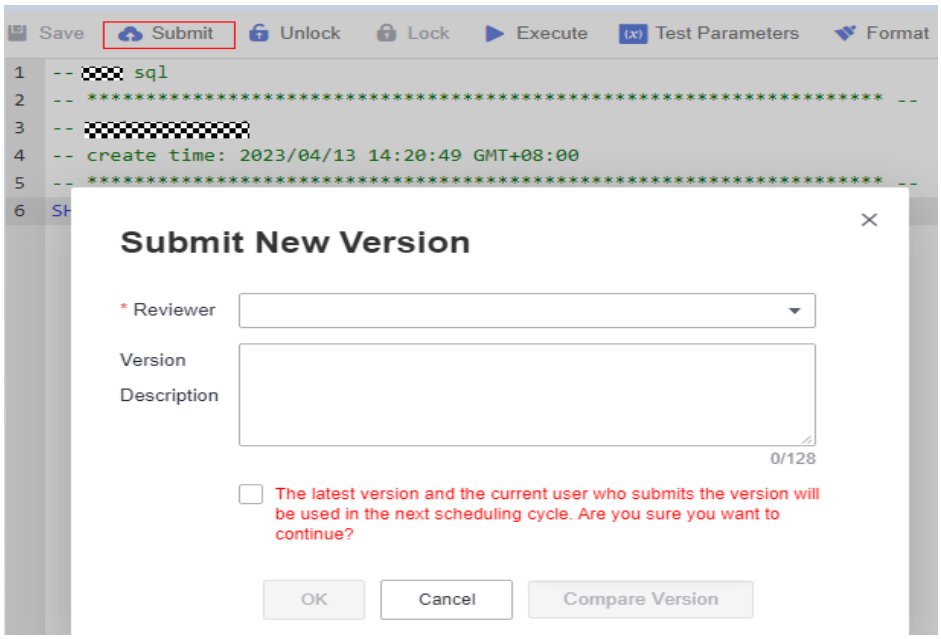
Step 2 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 3 In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.

Step 4 In the script directory, double-click the developed script to access the script development page.

Step 5 Above the script editor, click **Submit** to submit a version. In the displayed dialog box, select the reviewer, enter the change description (a maximum of 128 characters allowed), and select the check box below. If you do not select this option, you cannot click **OK**. When submitting a version, you can click **Compare Version** to view the differences between the current version and the last version.

Figure 6-6 Submitting a version



NOTE

- If review is enabled on the **Review Center** page, your submitted version will be reviewed by the workspace admin on the **Pending Review** tab page on the **Review Center** page. The version is submitted successfully only after it is approved by the admin. For details, see **Approval Settings**.

  To revoke a submitted request, go to the **Review Center** page and click the **My Applications** tab. Then you can submit an application again.
- If review is enabled, the following operations need to be reviewed: submitting scripts, deleting scripts, and importing submitted scripts.
- Before disabling the review function, ensure that there are no requests pending review in the current workspace.
- The enterprise mode does not support the review function.

----End

Version Rollback

After submitting the version, you can view it in the version list. (Currently, a maximum of 100 latest versions are saved.) Click **Roll Back** to roll back to any submitted version.

The rollback involves the following contents:

- DLI: data connections, databases, resource queues, and script contents
- DWS: data connections, databases, and script contents
- HIVE: data connections, databases, resource queues, and script contents
- SPARK: data connections, databases, and script contents
- SHELL: host connections, parameters, interactive parameters, and script contents
- RDS: data connections, databases, and script contents

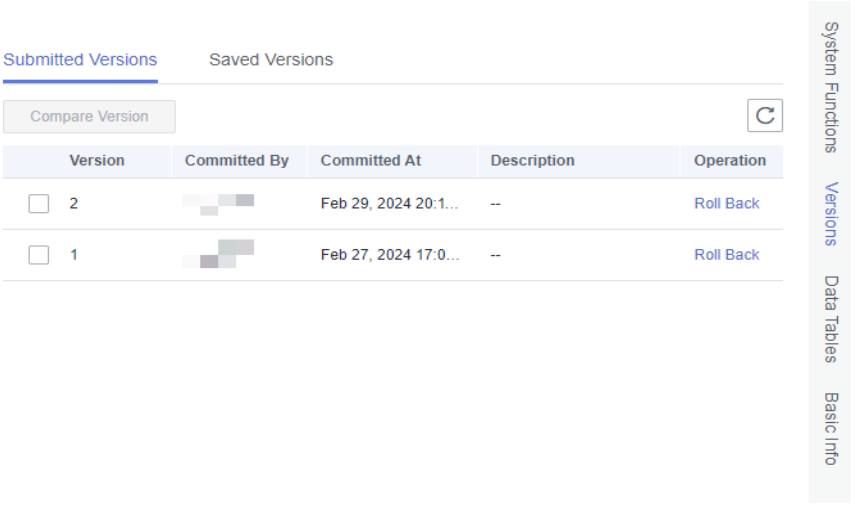
- PRESTO: data connections, modes, and script contents
- PYTHON: host connections, parameters, interactive parameters, and script content
- FLINK: script content

The procedure is as follows:

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. In the script directory list, double-click a script that you want to develop. The script development page is displayed.
5. On the right of the page, click the **Versions** tab and view the version submission records. Select the version to be rolled back and click **Roll Back**.

   If the content in the development state is not submitted, the content will be overwritten after the rollback. In this case, you must submit the rollback version again to make it take effect. By default, the latest submitted version is used for scheduling.

   Figure 6-7 Rolling back a version



Version Comparison

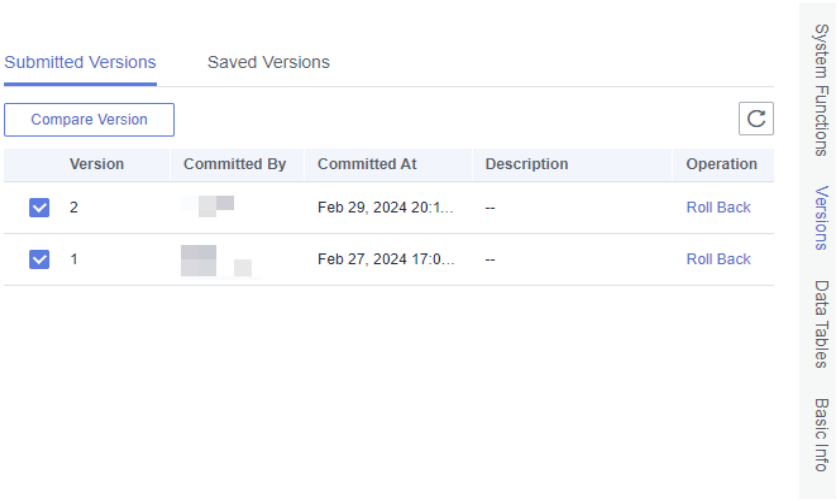
You can compare the script contents of two different versions. If you select only one version, the system compares the script content of the selected version with that in the development state. If you select two versions, the system compares the script contents of two different versions.

The procedure is as follows:

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

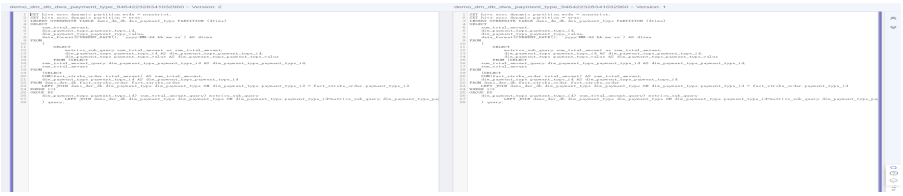
2. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
3. In the script directory list, double-click a script that you want to develop. The script development page is displayed.
4. On the right of the page, click the **Versions** tab and view the version submission records. Select the versions to be compared and click **Compare Version**.

Figure 6-8 Comparing versions



5. A new page is displayed, showing the script content of different versions on the left and right separately. The differences between the two versions have been marked. You can use the and buttons in the upper right corner to go to the previous or next change.

Figure 6-9 Version comparison details



6.3.5 Releasing a Script Task

In enterprise mode, when a developer submits a script version, the system generates a script release task. After the developer confirms releasing a package and the admin, deployer, a user with the DAYU Administrator or Tenant Administrator permission approves the package release request, the modified script is synchronized to the production environment.

NOTICE

- If the admin imported a submitted script, a release task will be generated.
- If the admin imported a released script, no release task will be generated.

Prerequisites

You have submitted a version. For details, see **Submitting a Version**.

Procedure

Step 1 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 2 In the left navigation pane, choose **Data Development** > **Task Release**.

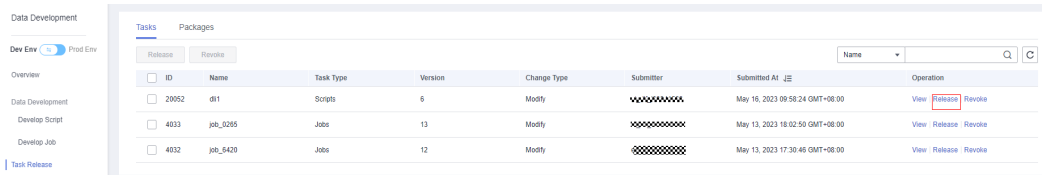
Step 3 On the **Tasks** page, the tasks generated for version submission are displayed. You can click **View** in the **Operation** column to view the modifications of a script compared with its previous version. After confirming that the modifications are correct, click **Release** to release the task.

You can filter tasks by name or submitter, and perform fuzzy search using a task name.

NOTE

- If you have only the developer permission, the script will be synchronized to the production environment only when the task is approved by the admin or deployer.
- After clicking **Release**, set the reviewer. The reviewer must be a workspace admin, deployer, or a user with the DAYU Administrator or Tenant Administrator permission. Set at least one reviewer and do not set yourself as the reviewer. Click **Reviewer Information** to go to the **Workspaces** page. Click **Edit** to configure reviewers.
- You can release a maximum of 100 tasks at a time. The tasks are released asynchronously. You can view the task release process.
- You can revoke tasks not to be released as a developer, deployer, or admin.

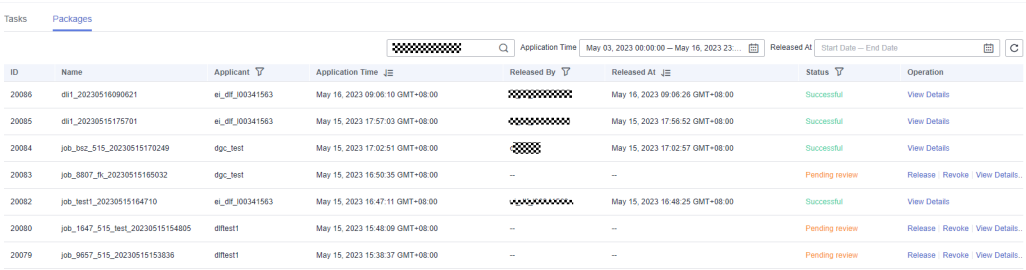
Figure 6-10 Clicking Release



Step 4 After the task is released, you can view the release status of the task on the **Packages** tab page. After approved, the task is released successfully.

You can filter packages by **Applicant**, **Application Time**, **Release At**, or **Released By**, and perform fuzzy search using a package name.

Figure 6-11 Viewing the task status

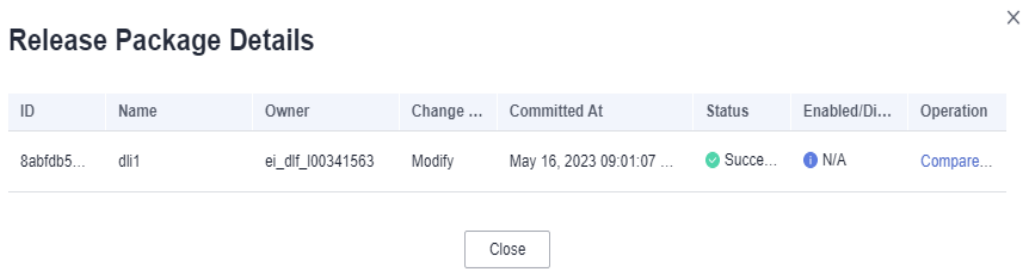


NOTE

You can revoke tasks not to be released as a developer, deployer, or admin.

After the task is released, you can click **View Details** in the **Operation** column to view the release status and startup status of the task. You can also click **Compare Version** in the **Operation** column to view the differences between different versions of release packages.

Figure 6-12 Viewing release package details



----End

6.3.6 (Optional) Managing Scripts

6.3.6.1 Copying a Script

This section describes how to copy a script.

Prerequisites

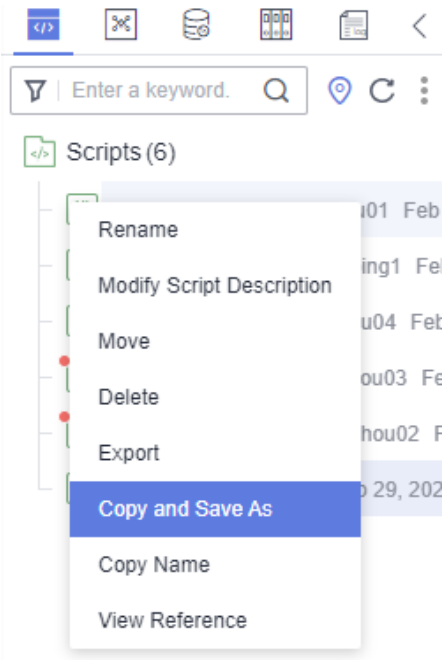
A script has been developed based on **Developing Scripts**.

Procedure

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.

4. In the script directory, select the script to be copied, right-click the script name, and choose **Copy Save As**.

Figure 6-13 Copying a script



5. In the displayed dialog box, configure related parameters. **Table 6-22** describes the parameters.

Table 6-22 Script directory parameters

Parameter	Description
Script Name	Name of the script. It contains a maximum of 128 characters. Only letters, digits, hyphens (-), underscores (_), and periods (.) are allowed. **NOTE** The name of the copied script cannot be the same as the name of the original script.
Select Directory	Parent directory of the script directory. The parent directory is the root directory by default.

6. Click **OK**.

6.3.6.2 Copying the Script Name and Renaming a Script

You can copy the name of a script and rename a script.

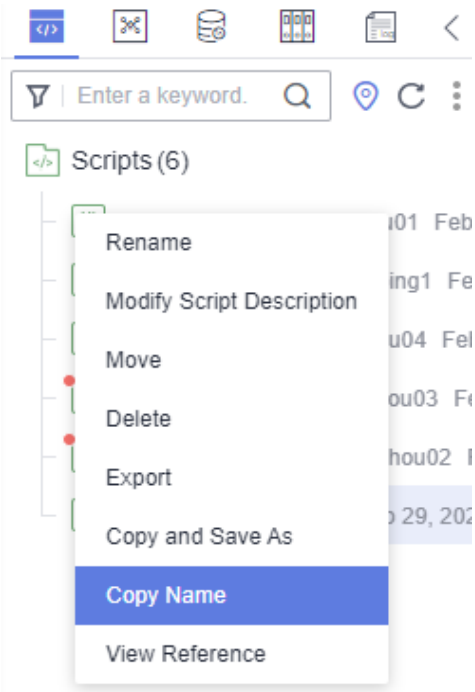
Prerequisites

A script has been developed based on **Developing Scripts**.

Copying the Script Name

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. Locate the target script in the script directory, right-click the script name, and select **Copy Name** to copy the script name to the clipboard.

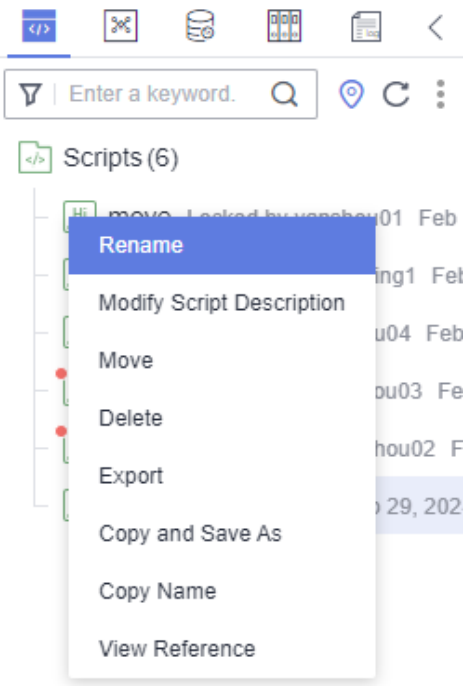
Figure 6-14 Copying the script name



Renaming a Script

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. Locate the target script In the script directory, right-click the script name, and select **Rename**.

Figure 6-15 Renaming a script

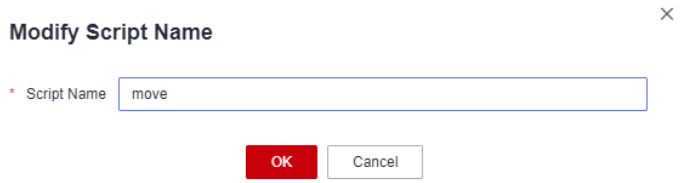


NOTE

An opened script file cannot be renamed.

5. In the displayed **Modify Script Name** dialog box, change the script name.

Figure 6-16 Renaming a script



6. Click **OK**.

6.3.6.3 Moving a Script or Script Directory

You can move a script file from one directory to another or move a script directory to another directory.

Prerequisites

A script has been developed based on **Developing Scripts**.

Procedure

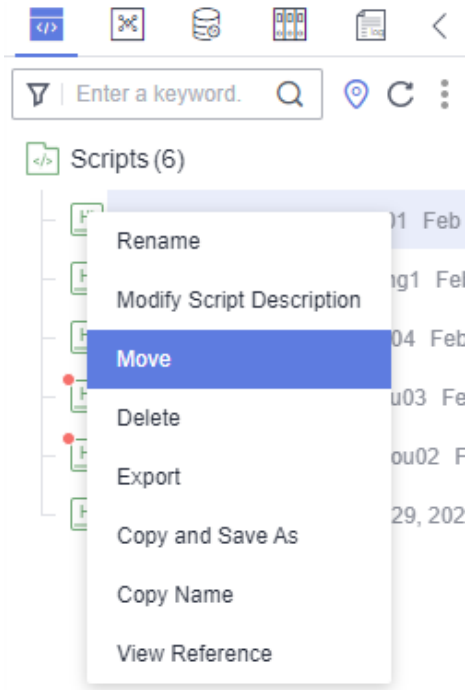
1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

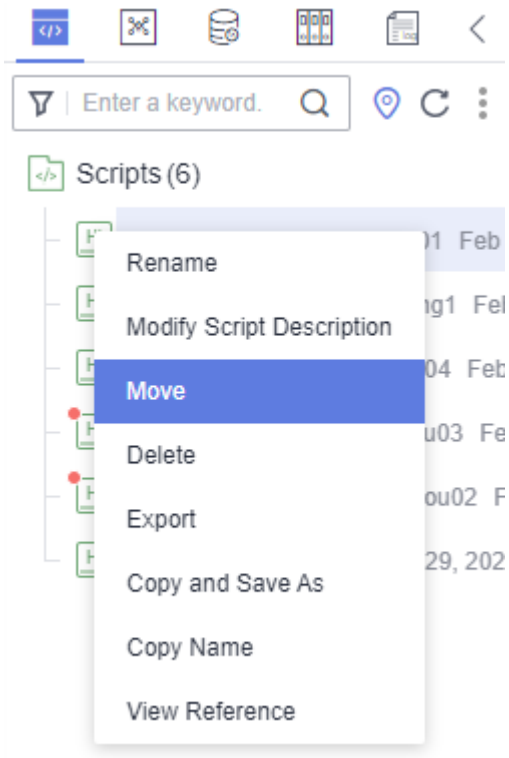
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. Move a script or script directory.

   Method 1: right-click

   a. In the script directory, right-click a script or script folder and select **Move**.

Figure 6-17 Selecting Move





b. In the displayed dialog box, configure related parameters. **Table 6-23** describes the parameters.

Figure 6-18 Moving a script

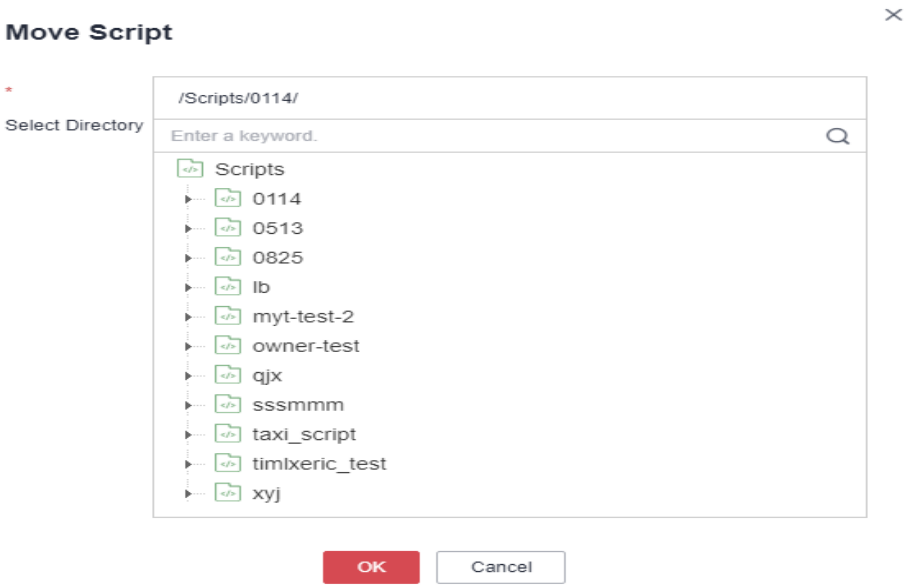


Figure 6-19 Move a directory

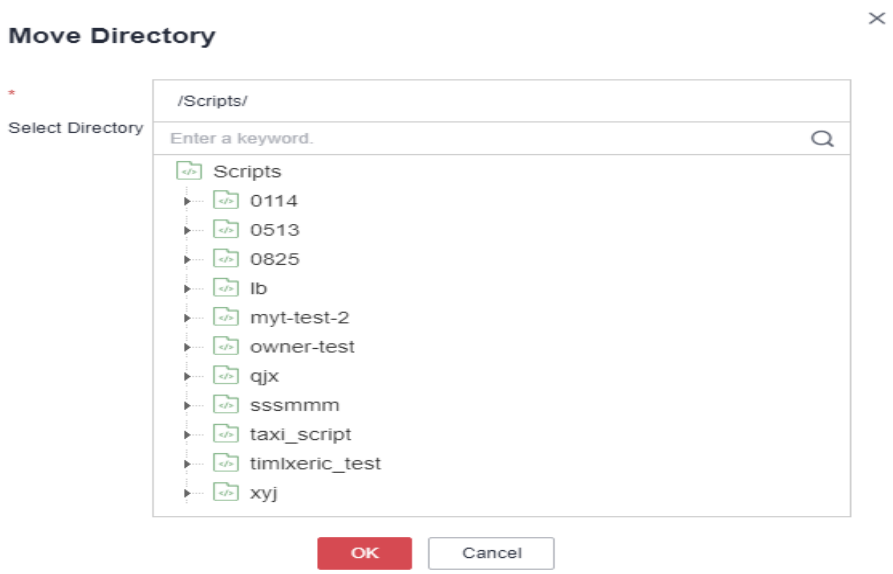


Table 6-23 Parameters for moving a script or directory

Parameter	Description
Select Directory	Directory to which the script or script directory is to be moved. The parent directory is the **root** directory by default.

c. Click **OK** to move the script or directory.

Method 2: drag-and-drop

Select a script or script folder and drag and drop it to the target folder.

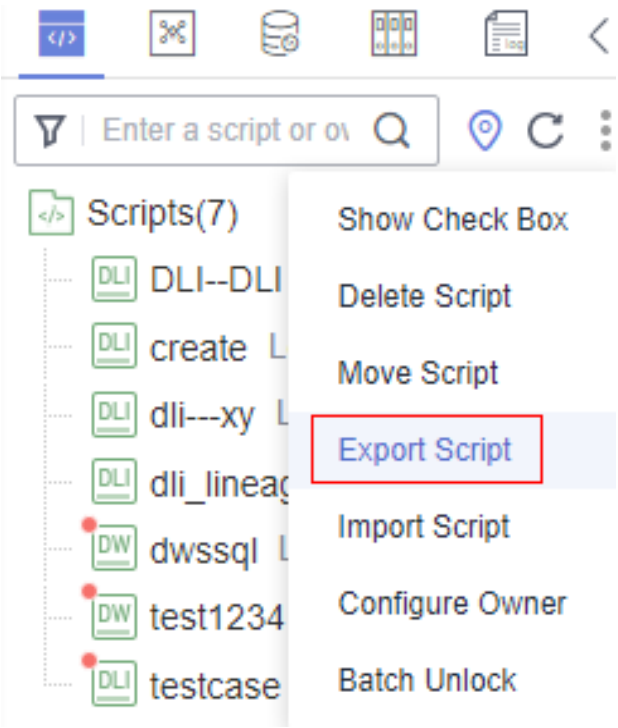
6.3.6.4 Exporting and Importing Scripts

Exporting Scripts

You can export one or more script files from the script directory. The exported files store the latest content in the development state.

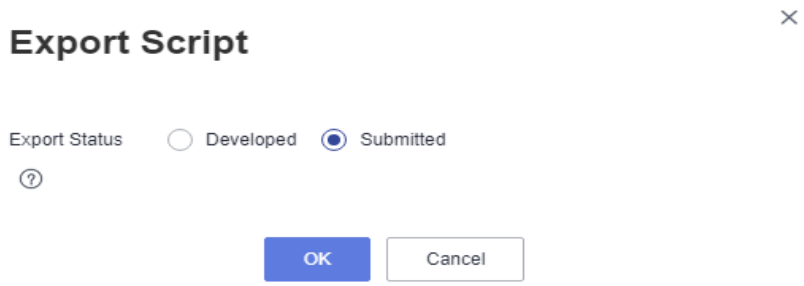
1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. Click in the script directory and select **Show Check Box**.
5. Select scripts, click , and select **Export Script**. After the export is successful, you can obtain the exported .zip file.

Figure 6-20 Selecting and exporting scripts



6. In the displayed **Export Script** dialog box, set **Export Status** and click **OK**.

Figure 6-21 Exporting scripts



Importing Scripts

This function is available only if the OBS service is available. If OBS is unavailable, scripts can be imported from the local PC.

You can import one or more script files in the script directory. After a job is imported, the content in the development state is overwritten and a new version is automatically submitted.

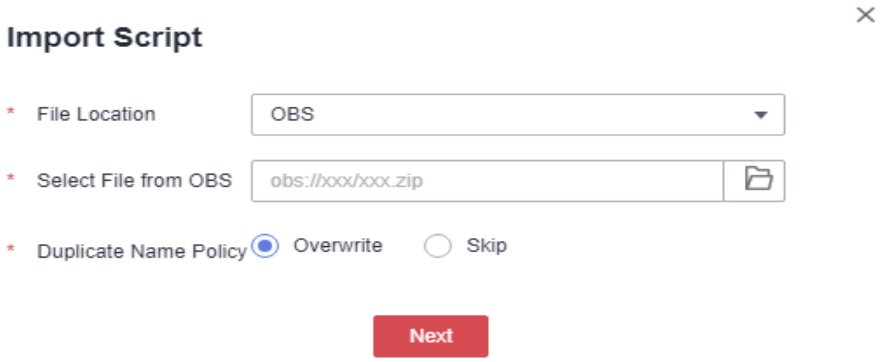
1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. Click and select **Import Script**. In the displayed dialog box, select the file to import and set **Duplicate Name Policy**.

   NOTE

   If you select **Overwrite** for **Duplicate Name Policy** but the hard lock policy is used and the script is locked by another user, the overwriting will fail. For details about soft and hard lock policies, see **Configuring the Hard and Soft Lock Policy**.

   Figure 6-22 Importing scripts

5. Click **Next**.

6.3.6.5 Viewing Script References

This section describes how to view the references of a script or all the scripts in a folder.

Prerequisites

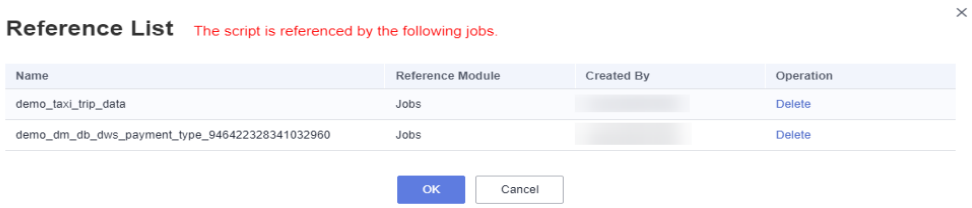
A script has been developed based on **Developing Scripts**.

Procedure

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. To view the references of a script, right-click the script and select **View Reference**.

   To view the references of all the scripts in a folder, right-click the folder and select **View Reference**.
5. In the displayed dialog box, you can view the references of a script or all the scripts in the folder.

Figure 6-23 References of a script



6.3.6.6 Deleting a Script

If you do not need to use a script any more, perform the following operations to delete it.

When you delete a script, the system checks whether the script is being referenced by some jobs. **Version** in the reference list lists the job versions that reference the script. After you click **Delete**, the job will be deleted as well as all version information about the job.

NOTE

If a script to be deleted is being associated with a job, ensure that services are not affected after the script is forcibly deleted. If you want to continue to use the job, go to the **Develop Job** page and associate the job with an available script.

Prerequisites

The script that you want to delete is not used by any jobs.

Deleting a Script

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. In the script directory, right-click the script that you want to delete and choose **Delete** from the shortcut menu.
5. In the displayed dialog box, click **OK**.

Batch Deleting Scripts

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. On the top of the script directory, click and select **Show Check Box**.

5. Select the scripts to be deleted, click , and select **Batch Delete**.
6. In the displayed dialog box, click **OK**.

6.3.6.7 Unlocking a Script

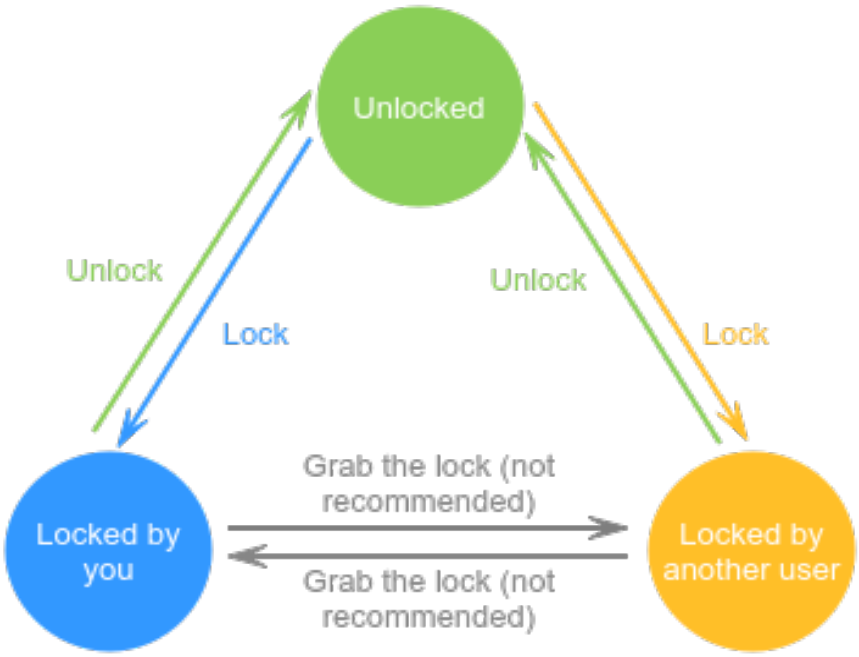
Script and job unlocking depends on the lock function of DataArts Factory.

The lock function prevents conflict caused by collaborative script or job development. If you create or import a script or job, it is locked by you by default. You can only edit, save, or submit a script or job you have locked. To edit, save, or submit a script or job that is locked by another user or not locked by any user, you must lock the script or job first.

NOTICE

- You can view the lock status of a script or job in the script or job directory tree.
- To view the latest version of an opened script or job locked by another user, you need to re-open the script or job because it is not updated in real time.
- Scripts or jobs that were created before the lock function was available are now unlocked by default. To edit, save, or submit these scripts or jobs, you need to lock them first.
- The locking operation depends on the soft and hard lock policies. For details about how to configure soft and hard lock policies, see **Configuring a Default Item**.
  - **Soft lock**: You can lock or unlock jobs or scripts, regardless of whether they are locked by others.
  - **Hard Lock**: You can lock jobs or scripts only after they have been unlocked by other users. The space administrator and the Administrator can lock and unlock jobs or scripts without any limitations.
- Do not lock a script or job that is locked by another user because if you do so, changes to the script or job made by the user will be lost. If you want to modify the script or job, contact the user to unlock the script or job, and lock it by yourself.

Figure 6-24 Lock statuses



Prerequisites

A script has been developed.

Unlocking a Script

If you submit a version, the latest script in the development state will be saved and submitted and overwrite the previous script version. You are advised to unlock the script after submitting the version so that other developers can modify the script as needed.

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

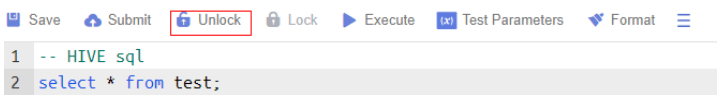
Step 2 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 3 In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.

Step 4 In the script directory, double-click the developed script to access the script development page.

Step 5 In the upper part of the script editor, click **Unlock** to unlock the script.

Figure 6-25 Unlocking a script



----End

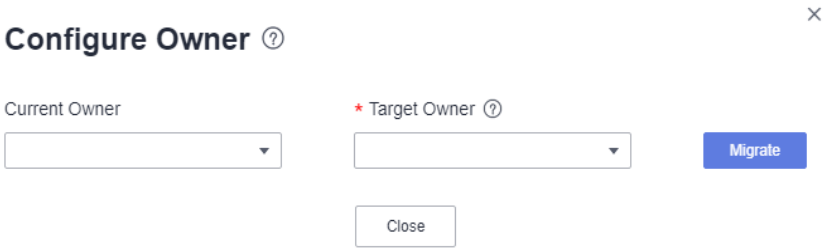
6.3.6.8 Changing the Script Owner

DataArts Factory allows you to change the script owner with a few clicks.

Procedure

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. At the top of the script directory, click and select **Configure Owner**.

   Figure 6-26 Changing the owner

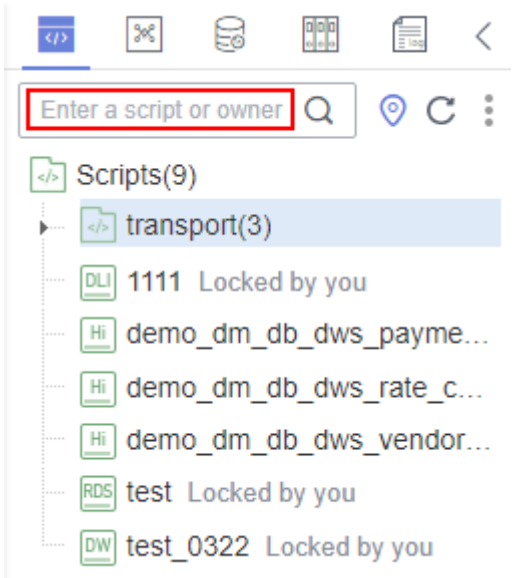
   

5. Set **Current Owner** and **Target Owner** and click **Migrate**.
6. When the owner is changed, click **Close**.

Related Operations

You can use an owner to filter scripts by entering the owner in the search box above the script directory.

Figure 6-27 Filtering scripts by owner



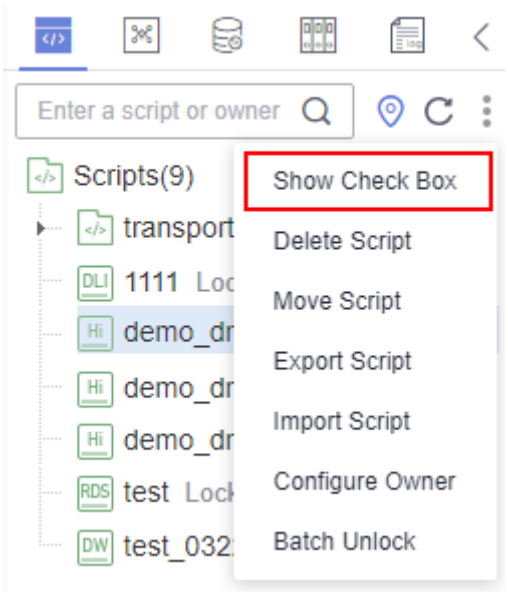
6.3.6.9 Unlocking Scripts

This section describes how to unlock scripts in batches.

Procedure

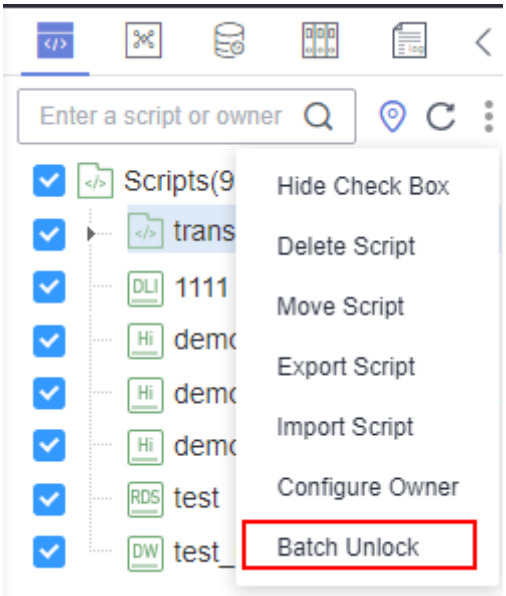
1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
4. Click in the script directory and select **Show Check Box**.

   Figure 6-28 Clicking Show Check Box

5. Select the scripts to unlock, click , and select **Batch Unlock**. The message "Unlocked." is displayed.

Figure 6-29 Batch Unlock



6.4 Job Development

6.4.1 Job Development Process

The job development function provides the following capabilities:

- Provides a graphical designer that allows you to quickly build a data processing workflow by drag-and-drop.
- Presets multiple job types, such as data integration, computing and analysis, data monitoring, and resource management, and completes complex data analysis and processing based on dependencies between jobs.
- Supports various scheduling modes.
- Supports job import and export.
- Monitors job status and sends job result notifications.
- Provides editing locks for collaborative development.
- Supports job version management and generation of saved and submitted versions.

  NOTE

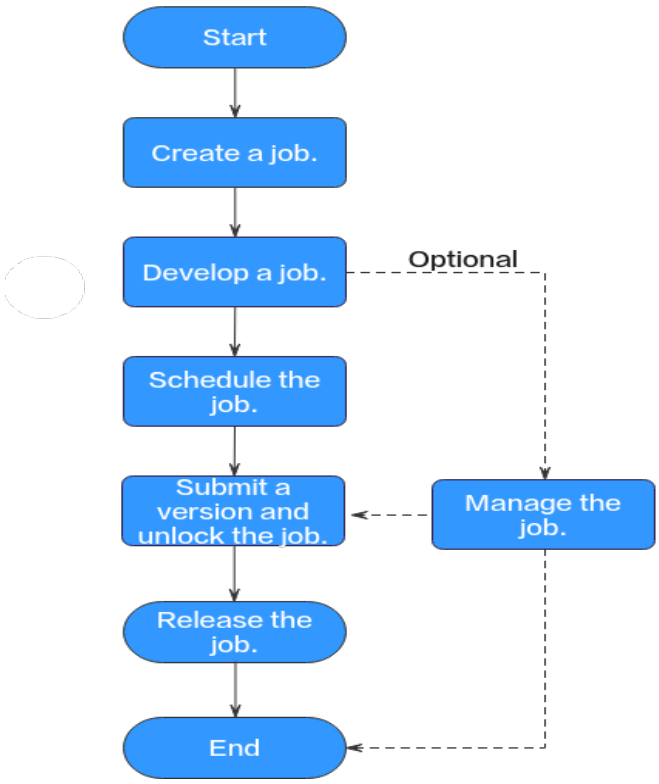
  If you save a script multiple times within a minute, only one version is recorded. If the intermediate data is important, you can click **Save new version** to save and add a version.

- Allows you to copy long job names. Click , perform fuzzy search to query matched scripts, and click the copy button next to a long job name to copy it.
- Allows you to right-click a job to quickly copy the job name and to quickly close an opened job tab page.
- Provides a link in the execution results of single-task MRS Spark SQL and MRS Hive SQL jobs that use a connection of the MRS API type. Through this link, you can switch to MRS Yarn to view execution logs.

- Allows you to switch to the task release page by clicking **Release** when developing a job in enterprise mode.
- Allows you to filter submitted, unsubmitted, scheduled, and unscheduled jobs. Unsubmitted jobs are marked in red, and unscheduled jobs are marked in yellow.
- Allows you to configure the SQL editor style for single-task jobs. Click **Style Configuration** to configure the editor, icon display, annotation templates, and shortcut keys that can be used in the SQL script editor.
- Allows you to view single-task SQL query results in a table or list. You can click **Style Configuration** and set **SQL Query Result Display Mode** on the **Configure Editor** tab page.

Before developing a job, you can learn about the basic job development process.

Figure 6-30 Job development process



1. Create a job: Currently, two job types are available: batch and real-time, which are used for batch data processing and real-time connection data processing, respectively. Batch jobs support pipeline and single-node modes. For details, see **Creating a Job**.
2. Develop the job: Develop the created job. You can orchestrate and configure nodes. For details, see **Developing a Pipeline Job**.
3. Schedule the job: Configure job scheduling tasks. For details, see **Setting Up Scheduling for a Job**.
   - If the processing mode of a job is batch processing, configure scheduling types for jobs. Three scheduling types are supported: run once, run periodically, and event-based. For details, see **Setting Up Scheduling for a Job Using the Batch Processing Mode**.

- If the processing mode of a job is real-time processing, configure scheduling types for nodes. Three scheduling types are supported: run once, run periodically, and event-based. For details, see **Setting Up Scheduling for Nodes of a Job Using the Real-Time Processing Mode**.

4. Submit a version and unlock the script: After performing this step, the job can be scheduled and modified by other developers. For details, see **Submitting a Version**.
5. (Optional) Manage the job: After the job development is complete, you can manage the job as required. For details, see **(Optional) Managing Jobs**.
6. Release the job. This step is required in enterprise mode. For details, see **Releasing a Job Task**.

6.4.2 Creating a Job

A job is composed of one or more nodes that are performed collaboratively to complete data operations. Before developing a job, create a new one.

Prerequisites

Each workspace can hold a maximum of 10,000 jobs. Ensure that the number of your jobs does not reach this upper limit.

(Optional) Creating a Directory

If a directory is available, skip this step.

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. In the job directory list, right-click a directory and choose **Create Directory** from the shortcut menu.
5. In the **Create Directory** dialog box, configure directory parameters based on **Table 6-24**.

Table 6-24 Job directory parameters

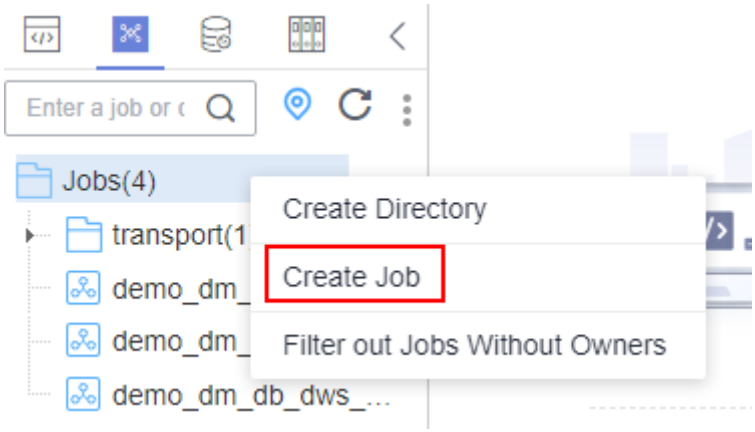
Parameter	Description
Directory Name	Name of a job directory. The name must contain 1 to 64 characters, including only letters, numbers, underscores (_), and hyphens (-).
Select Directory	Parent directory of the job directory. The parent directory is the root directory by default.

6. Click **OK**.

Creating a Job

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. In the job directory list, right-click a directory and select **Create Directory**.

Figure 6-31 Creating a job



5. In the displayed dialog box, configure job parameters. **Table 6-25** describes the job parameters.

Table 6-25 Job parameters

Parameter	Description
Job Name	Name of the job. The name must contain 1 to 128 characters, including only letters, numbers, hyphens (-), underscores (_), and periods (.).

Parameter	Description
Processing Mode	Type of the job. • **Batch processing**: Data is processed periodically in batches based on the scheduling plan, which is used in scenarios with low real-time requirements. This type of job is a pipeline that consists of one or more nodes and is scheduled as a whole. It cannot run for an unlimited period of time, that is, it must end after running for a certain period of time. You can configure job-level scheduling tasks for this type of job. That is, the job is scheduled as a whole. For details, see **Setting Up Scheduling for a Job Using the Batch Processing Mode**. • **Real-time processing**: Data is processed in real time, which is used in scenarios with high real-time performance. This type of job is a business relationship that consists of one or more nodes. You can configure scheduling policies for each nodes, and the tasks started by nodes can keep running for an unlimited period of time. In this type of job, lines with arrows represent only service relationships, rather than task execution processes or data flows. You can configure node-level scheduling tasks for this type of job, that is, each node can be independently scheduled. For details, see **Setting Up Scheduling for Nodes of a Job Using the Real-Time Processing Mode**.
Mode	• **Pipeline**: You drag and drop one or more nodes to the canvas to create a job. The nodes are executed in sequence like a pipeline. **NOTE** In enterprise mode, real-time processing jobs do not support the pipeline mode. • **Single task**: The job contains only one node. Currently, this mode supports DLI SQL, DWS SQL, RDS SQL, MRS Hive SQL, MRS Spark SQL, Doris SQL, DLI Spark, Flink SQL, and Flink JAR nodes. Instead of creating a script and referencing the script in the node of a job, you can debug the script and configure scheduling in the SQL editor of a single-task job. **NOTE** Currently, jobs with a single Flink SQL node support MRS 3.2.0-LTS.1 and later versions.
Migration Type	This parameter is available when **Mode** is **Single task Data Migration**. The default value is **Table/File Migration**.
Select Directory	Directory to which the job belongs. The root directory is selected by default.
Owner	Owner of the job.

Parameter	Description
Priority	Priority of the job. The value can be **High**, **Medium**, or **Low**. **NOTE** Job priority is a label attribute of the job and does not affect the scheduling and execution sequence of the job.
Agency	After an agency is configured, the job interacts with other services as an agency during job execution. If an agency has been configured for the workspace (for details, see **Configuring a Public Agency**), the job uses the workspace-level agency by default. You can also change the agency to a job-level agency by referring to **Configuring a Job-Level Agency**. **NOTE** Job-level agency takes precedence over workspace-level agency.
Log Path	Selects the OBS path to save job logs. By default, logs are stored in a bucket named **dlf-log-**{*Projectid*}. **NOTE** • If you want to customize a storage path, select the bucket that you have created on OBS by following the instructions provided in **(Optional) Changing a Job Log Storage Path**. • Ensure that you have the read and write permissions on the OBS path specified by this parameter. Otherwise, the system cannot write logs or display logs.

6. Click **OK**.

6.4.3 Developing a Pipeline Job

This section describes how to develop and configure a job.

For details about how to develop a batch processing job or real-time processing job in pipeline mode, see **Compiling Job Nodes**, **Configuring Basic Job Information**, **Configuring Job Parameters**, and **Testing and Saving the Job**.

Prerequisites

- A job has been created. For details, see **Creating a Job**.
- You have locked the job. Otherwise, you must click **Lock** so that you can develop the job. A job you create or import is locked by you by default. For details, see the **lock function**.

Compiling Job Nodes

This part applies to batch processing jobs and real-time processing jobs in pipeline mode.

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. In the job directory, double-click the name of a batch processing job or real-time processing job in pipeline mode to access the job development page.
5. Drag a desired node to the canvas, move the mouse over the node, and select the icon and drag it to connect to another node.

   NOTE

   It is recommended that each job contain a maximum of 200 nodes.

Figure 6-32 Compiling a job

6. Configure node functions. Right-click a node icon on the canvas and select a function as needed. **Table 6-26** lists the available functions.

Table 6-26 Node functions

Function	Description
Configure	Goes to the **Node Property** page of the node.
Delete	Deletes one or more nodes at the same time. • Deleting one node: Right-click the node icon in the canvas and choose **Delete** or press the **Delete** shortcut key. • Deleting multiple nodes: Click the icons of the nodes to be deleted in the canvas while holding on **Ctrl**, right-click the blank area of the current job canvas, and choose **Delete** or press the **Delete** shortcut key.

Function	Description
Copy	Copies one or more nodes to any job. • Single-node copy: You can either right-click the node icon in the canvas, choose **Copy**, and paste the node to a target location, or click the node icon in the canvas and press **Ctrl +C** and **Ctrl+V** to paste the node to a target location. The copied node carries the configuration information of the original node. • Multi-node copy: Click the icons of the nodes to be copied in the canvas while holding on **Ctrl**. Then you can either right-click the blank area of the canvas, choose **Copy**, and paste the nodes to a target location, or press **Ctrl+C** and **Ctrl+V** to paste the nodes to a target location. The copied node carries the configuration information of the original node, but does not contain the connection relationship between nodes.
Test Run	Runs the node for a test. **NOTE** You can view the test run logs of the job node by clicking **View Log**.
Test from Current Node	This option is available only for batch processing jobs. It tests the current and subsequent nodes.
Add/ Delete Connection	Adds or deletes a connection between two nodes.
Edit CDM Job	This option is available only for CDM jobs. After selecting a CDM cluster and a job, you can go to the CDM job editing page to modify the job.
View Job Log	This option is available only for CDM jobs. When a CDM job is running, you can right-click the CDM job node and select **View Job Log** from the shortcut menu to go to the job monitoring page and view logs to help developers demarcate and locate job running exceptions.
Edit Script	This option is available only for the node associated with a script. Goes to the script editing page and edits the associated script.
Add Note	Adds a note to the node. Each node can have multiple notes.

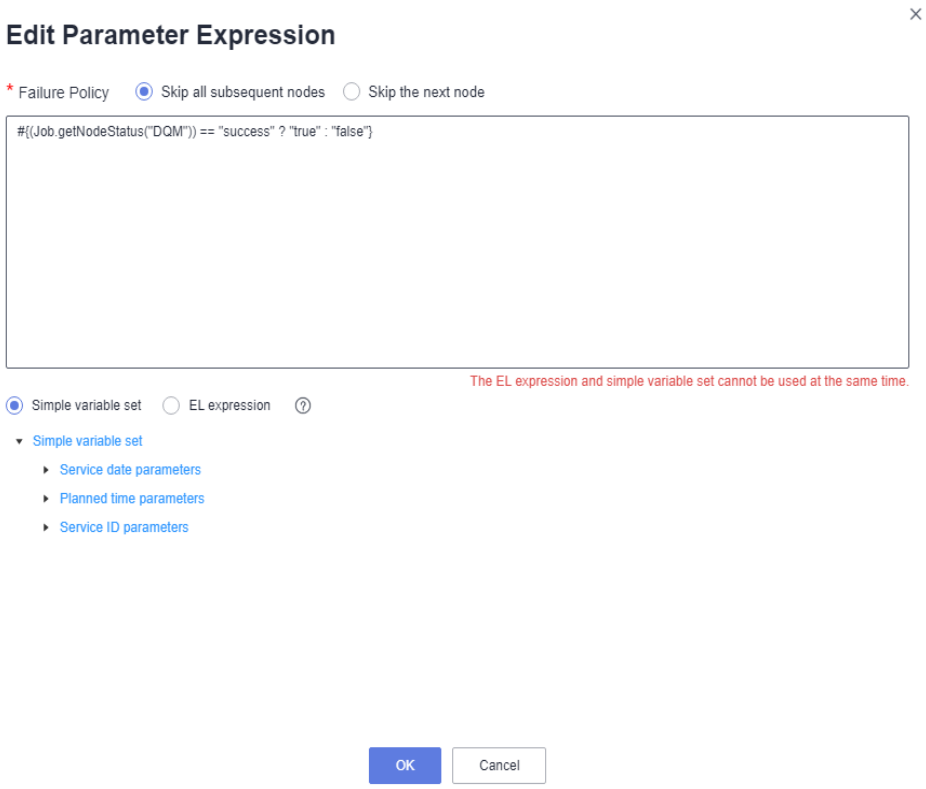
7. (Optional) Configure line functions. Right-click the line connecting two nodes on the canvas. **Delete** and **Set Condition** are displayed. You can select them as needed.
   - **Delete**: Deletes the line connecting the nodes.
   - **Set Condition**: In the displayed dialog box, you can enter a ternary expression using the EL expression syntax. If the result of the ternary

expression is **true**, subsequent nodes will be connected. Otherwise, subsequent nodes will be skipped.

The following figure shows a typical ternary expression. If the execution result of the DQM node is **true**, subsequent nodes will be connected. If the execution result is **false** and the **Failure Policy** is **Skip all subsequent nodes**, the next node A and all nodes following node A will be skipped.

```
#{(Job.getNodeStatus("DQM")) == "success" ? "true" : "false"}
```

Figure 6-33 Set Condition



For details about the EL expression syntax, see **Expression Overview**. For details about how to use IF conditions, see **IF Condition Judgment**.

8. Configure node properties Click a node in the canvas. On the displayed **Node Properties** page, configure node properties. For details, see **Node Overview**.

Configuring Basic Job Information

After you configure the owner and priority for a job, you can search for the job by the owner and priority. The procedure is as follows:

Click the **Basic Info** tab on the right of the canvas to expand the configuration page and configure job parameters, as listed in **Table 6-27**.

Table 6-27 Basic job information

Parameter	Description
Owner	An owner configured during job creation is automatically matched. This parameter value can be modified.
Executor	This parameter is available when **Scheduling Identities** is set to **Yes**. User that executes the job. When you enter an executor, the job is executed by the executor. If the executor is left unspecified, the job is executed by the user who submitted the job for startup.
Job Agency	This parameter is available when **Scheduling Identities** is set to **Yes**. After an agency is configured, the job interacts with other services as an agency during job execution.
Priority	Priority configured during job creation is automatically matched. This parameter value can be modified.
Execution Timeout	Timeout of the job instance. If this parameter is set to 0 or is not set, this parameter does not take effect. If the notification function is enabled for the job and the execution time of the job instance exceeds the preset value, the system sends a specified notification, and the job keeps running.
Exclude Waiting Time from Instance Timeout Duration	Whether to exclude the wait time from the instance execution timeout duration If you select this option, the time to wait before an instance starts running is excluded from the timeout duration. You can modify this setting on the **Default Configuration** page. If you do not select this option, the time to wait before an instance starts running is included in the timeout duration.
Custom Parameter	Set the name and value of the parameter.
Job Tag	Configure job tags to manage jobs by category. Click **Add** to add a tag to the job. You can also select a tag configured in **Managing Job Tags**.

Configuring Job Parameters

Job parameters can be globally used in any node in jobs. The procedure is as follows:

For batch and real-time processing jobs in pipeline mode: Click the blank area in the canvas and then the **Parameter Setup** tab on the right, and configure the parameters listed in **Table 6-28**.

Table 6-28 Job parameter setup

Function	Description
Variables	
Add	Click **Add** and enter the variable parameter name and parameter value in the text boxes. • Parameter Name Only letters, numbers, hyphens, and underscores (_) are allowed. • Parameter Value – The string type of parameter value is a character string, for example, **str1**. – The numeric type of parameter value is a number or operation expression. After the parameter is configured, it is referenced in the format of **$**{*parameter name*} in the job. **NOTE** If a job has two nodes, the first Rest Client node returns a body, and the second node uses the returned data. If the data contains more than 1,000,000 characters, it will be truncated. When configuring job parameters, ensure that the value of a job parameter contains no more than 1,000,000 characters.
Edit Parameter Expression	Click  next to the parameter value text box. In the displayed dialog box, edit the parameter expression. For more expressions, see **Expression Overview**.
Modify	Change the parameter name or value in the corresponding text boxes.
Mask	If the parameter value is a key, click  to mask the value for security purposes.
Delete	Click next to the parameter name and value text boxes to delete the job parameter.
Constant Parameter	

Function	Description
Add	Click **Add** and enter the constant parameter name and parameter value in the text boxes. • Parameter name Only letters, numbers, hyphens, and underscores (_) are allowed. • Parameter value – The string type of parameter value is a character string, for example, **str1**. – The numeric type of parameter value is a number or operation expression. After the parameter is configured, it is referenced in the format of **$**{*parameter name*} in the job.
Edit Parameter Expression	Click  next to the parameter value text box. In the displayed dialog box, edit the parameter expression. For more expressions, see **Expression Overview**.
Modify	Modify the parameter name and parameter value in text boxes and save the modifications.
Delete	Click next to the parameter name and value text boxes to delete the job parameter.
Workspace Environment Variables	
View the variables and constants that have been configured in the workspace.	

Click the **Parameter Preview** tab and configure the parameters listed in **Table 6-29**.

NOTE

The script parameters of the following types of operators can be previewed: MRS Flink Job, DLI Flink Job, DLI SQL, DWS SQL, MRS HetuEngine, MRS ClickHouse SQL, MRS Hive SQL, MRS Impala SQL, MRS Presto SQL, RDS SQL, and MRS Spark SQL.

Table 6-29 Job parameter preview

Function	Description
Current Time	This parameter is displayed only when **Scheduling Type** is set to **Run once**. The default value is the current time.
Event Triggering Time	This parameter is displayed only when **Scheduling Type** is set to **Event-based**. The default value is the time when an event is triggered.

Function	Description
Scheduling Period	This parameter is displayed only when **Scheduling Type** is set to **Run periodically**. The default value is the scheduling period.
Start Time	This parameter is displayed only when **Scheduling Type** is set to **Run periodically**. The value is the configured job execution time.
Start Time	This parameter is displayed only when **Scheduling Type** is set to **Run periodically**. The value is the time when the periodic job scheduling starts.
Subsequent Instances	Number of job instances scheduled. • The default value is **1** when **Scheduling Type** is set to **Run once**. • The default value is **1** when **Scheduling Type** is set to **Event-based**. • When **Scheduling Type** is set to **Run periodically**: If the number of instances exceeds 10, a maximum of 10 instances can be displayed, and the system displays message "A maximum of 10 instances are supported."

NOTE

In **Parameter Preview**, if a job parameter has a syntax error, the system displays a message.

If a parameter depends on the data generated during job execution, such data cannot be simulated and displayed in **Parameter Preview**.

Testing and Saving the Job

After a job is configured, complete the following operations:

Batch processing job

Step 1 Click ▶ to test the job. If the test fails, view the logs of the job node and locate and rectify the fault.

NOTE

You can view the test run logs of the job by clicking **View Log**.

If you test the job before submitting a version, the version of the generated job instance is 0 on the **Job Monitoring** page.

Step 2 When the test is successful, click 💾 to save the job configuration.

After the job is saved, a version is automatically generated and displayed in **Versions**. The version can be rolled back. If you save a job multiple times within a minute, only one version is recorded. If the intermediate data is important, you can click **Save new version** to save and add a version.

----End

Processing jobs in real time

Step 1 Click to save the job configuration.

After the job is saved, a version is automatically generated and displayed in **Versions**. The version can be rolled back. If you save a job multiple times within a minute, only one version is recorded. If the intermediate data is important, you can click **Save new version** to save and add a version.

----End

6.4.4 Developing a Batch Processing Single-Task SQL Job

This section describes how to develop and configure a job.

For details about how to develop a batch processing job in single-task mode, see sections **Developing an SQL Script**, **Configuring job parameters**, **Monitoring Quality**, **Data Table**, **Testing and Saving the Job**, and **Downloading or Dumping a Script Execution Result**.

Prerequisites

- A job has been created. For details, see **Creating a Job**.
- You have locked the job. Otherwise, you must click **Lock** so that you can develop the job. A job you create or import is locked by you by default. For details, see the **lock function**.

Developing an SQL Script

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. In the job directory, double-click the name of a single-task job to access the job development page.
5. On the right of the SQL editor, click **Basic Info** to configure basic information, properties, and advanced settings of the job. **Table 6-30** lists the basic information, **Table 6-31** lists the properties, and **Table 6-32** lists the advanced settings.

Table 6-30 Basic job information

Parameter	Description
Owner	An owner configured during job creation is automatically matched. This parameter value can be modified.

Parameter	Description
Executor	This parameter is available when **Scheduling Identities** is set to **Yes**. User that executes the job. When you enter an executor, the job is executed by the executor. If the executor is left unspecified, the job is executed by the user who submitted the job for startup.
Job Agency	This parameter is available when **Scheduling Identities** is set to **Yes**. After an agency is configured, the job interacts with other services as an agency during job execution.
Priority	Priority configured during job creation is automatically matched. This parameter value can be modified.
Execution Timeout	Timeout of the job instance. If this parameter is set to 0 or is not set, this parameter does not take effect. If the notification function is enabled for the job and the execution time of the job instance exceeds the preset value, the system sends a specified notification, and the job keeps running.
Exclude Waiting Time from Instance Timeout Duration	Whether to exclude the wait time from the instance execution timeout duration If you select this option, the time to wait before an instance starts running is excluded from the timeout duration. You can modify this setting on the **Default Configuration** page. If you do not select this option, the time to wait before an instance starts running is included in the timeout duration.
Custom Parameter	Set the name and value of the parameter.
Job Tag	Configure job tags to manage jobs by category. Click **Add** to add a tag to the job. You can also select a tag configured in **Managing Job Tags**.

Table 6-31 Properties of a single-task job

Property	Description
DLI SQL properties	
DLI Data Directory	Select the DLI data directory. • Default DLI data directory **dli** • Metadata catalog that has been created in LakeFormation associated with DLI.

Property	Description
Database Name	Select a database. If you select the default DLI data directory **dli**, select a DLI database and tables. If you select a metadata catalog that has been created in LakeFormation associated with DLI, select a LakeFormation database and tables.
Queue Name	The queue set in the SQL script is selected by default. You can change another one. You can create a resource queue using either of the following methods: ● Click  . On the displayed **Queue Management** page of DLI, create a resource queue. ● Go to the DLI console to create a resource queue.
Record Dirty Data	Click  to specify whether to record dirty data. ● If you select , dirty data will be recorded. ● If you do not select , dirty data will not be recorded.

Property	Description
DLI Environmental Variable	• The environment variable must start with **dli.sql.** or **spark.sql.** • If the key of the environment variable is **dli.sql.shuffle.partitions** or **dli.sql.autoBroadcastJoinThreshold**, the environment variable cannot contain the greater than (>) or less than (<) sign. • If a parameter with the same name is configured in both a job and a script, the parameter value configured in the job will overwrite that configured in the script. **NOTE** When you run a DLI SQL script or test a DLI SQL single-task job in non-scheduling scenarios, the following parameters are enabled by default: • **spark.sql.adaptive.enabled**: Adaptive Query Execution (AQE) is enabled so that Spark can dynamically optimize the query execution plan based on the characteristics of the data being processed and improve the performance by reducing the amount of data to be processed. • **spark.sql.adaptive.join.enabled**: AQE is enabled for join operations. The optimal join algorithm is selected based on the data being processed to improve performance. • **spark.sql.adaptive.skewedJoin.enabled**: AQE is enabled for skewed join operations. Skewed data can be automatically detected and the join algorithm is optimized accordingly to improve performance. • **spark.sql.mergeSmallFiles.enabled**: Merging of small files is enabled. Small files can be merged into large ones, improving performance and shortening the processing time. In addition, less files need to be read from remote storage, and more local files can be used. If you do not want to use these functions, you can set the values of the preceding parameters to **false**.
DWS SQL properties	
Data Connection	Select a data connection.
Database	Select a database.
Dirty Data Table	Name of the dirty data table defined in the SQL script. The dirty data attributes cannot be edited. They are automatically recommended by the SQL script content.
Matching Rule	Enter a Java regular expression used to match the DWS SQL result. For example, if the expression is (?<=\()(-*\d+?)(?=,) and the SQL result is (1,"error message"), then the matched result is "1".
Failure Matching Value	If the matched content equals the set value, the node fails to be executed.

Property	Description
RDS SQL properties	
Data Connection	Select a data connection.
Database	Select a database.
Spark SQL properties	
MRS Job Name	MRS job name. The system automatically sets this parameter based on the job name. If the MRS job name is not set and the direct connection mode is selected, the node name can contain only letters, digits, hyphens (-), and underscores (_). A maximum of 64 characters are allowed, and Chinese characters are not allowed.
Data Connection	Select a data connection.
MRS Resource Queue	Select a created MRS resource queue.
Database	Select a database.
Program Parameter	Set program parameters. The following is an example: Set **Parameter** to **--queue** and **Value** to **default_cr**, indicating that a specified queue of the MRS cluster is configured. You can also go to the MRS console, click the name of the MRS cluster and then the **Jobs** tab, locate the job, click **More** in the **Operation** column, and select View **Details** to view the job details. **NOTE** Configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. This configuration is unavailable if a Spark proxy connection is used. Spark SQL jobs with a single operator and using a connection of the MRS API type support program parameters.
Hive SQL properties	
MRS Job Name	MRS job name. The system automatically sets this parameter based on the job name. If the MRS job name is not set and the direct connection mode is selected, the node name can contain only letters, digits, hyphens (-), and underscores (_). A maximum of 64 characters are allowed, and Chinese characters are not allowed.

Property	Description
Data Connection	Select a data connection.
Database	Select a database.
MRS Resource Queue	Select a created MRS resource queue.
Program Parameter	Set program parameters. The following is an example: Set **Parameter** to **--hiveconf** and **Value** to **mapreduce.job.queuename=default_cr**, indicating that a specified queue of the MRS cluster is configured. You can also go to the MRS console, click the name of the MRS cluster and then the **Jobs** tab, locate the job, click **More** in the **Operation** column, and select View **Details** to view the job details. **NOTE** Configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. This configuration is unavailable if a Hive proxy connection is used. Hive SQL jobs with a single operator and using a connection of the MRS API type support program parameters.

Table 6-32 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system checks whether the node execution is complete. The value ranges from 1 to 60 seconds. During the node execution, the system checks whether the node execution is complete at the configured interval.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. • **Yes**: The node will be re-executed, and the following parameters must be configured: – Retry upon Timeout – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default value. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current node Fails	Yes	Policy for handling subsequent nodes if the current node fails • **End the current job execution plan**: Execution of the current job will stop, and the job instance status will become **Failed**. • **Ignore the failure and set the job execution result to success**: The failure of the current node will be ignored, and the next node will be executed. The job instance status will become **Successful**.

6. Enter one or more SQL statements in the SQL editor.

NOTE

- SQL statements are separated by semicolons (;). If semicolons are used in other places but not used to separate SQL statements, escape them with backslashes (\). The following is an example:
```
select 1;
select * from a where b="dsfa\;";  --example 1\;example 2.
```
- RDS SQL does not support the begin ... commit transaction syntax. If necessary, use the start transaction ... commit transaction syntax.
- The script cannot be larger than 16 MB.
- The system date obtained by using an SQL statement is different from that obtained by using the database tool. The query result is stored in the database in the YYYY-MM-DD format, but the query result displayed on the page is in the converted format.
- When a user submits a Spark SQL script to MRS, the script is submitted to the tenant queue bound to the user by default. The bound queue is the queue corresponding to tenant role of the user. If there are multiple queues, the system preferentially selects a queue based on the queue priorities. To set a fixed queue for the user to submit scripts, log in to FusionInsight Manager, choose **Tenant Resources** > **Dynamic Resource Plan**, and click the **Global User Policy** tab. For details, see **Managing Global User Policies**.

To facilitate script development, DataArts Factory provides the following capabilities:

– The script editor supports the following shortcut keys, which improve the script development efficiency:
  - **F8**: Run a script.
  - **F9**: Stop running a script.
  - **Ctrl** + **/**: Comment out or uncomment the line or code block at the cursor.
  - **Ctrl** +**Z**: Cancel
  - **Ctrl** + **F**: Search
  - **Ctrl** + **Shift** + **R**: Replace
  - **Ctrl** + **X**: Cut
  - **Ctrl** + **S**: Save a script.
  - **Alt** + mouse dragging: Select columns to edit a block.
  - **Ctrl** + mouse click: Select multiple lines to edit or indent them together.
  - **Shift** + **Ctrl** + **K**: Delete the current line.
  - **Ctrl** + → (or ←): Move the cursor rightwards (or leftwards) by word.
  - **Ctrl** + **Home** or **Ctrl** + **End**: Navigate to the beginning or end of the current file.
  - **Home** or **End**: Navigate to the beginning or end of the current line.

- **Ctrl** + **Shift** + **L**: Double-click all the same character strings and add cursors to them to implement batch modification.
- **Ctrl** + **D**: Delete a line.
- **Shift** + **Ctrl** + **U**: Unlock a script.
- **Ctrl** + **Alt** + **K**: Select the word where the cursor resides.
- **Ctrl** + **B**: Format
- **Ctrl** + **Shift** + **Z**: Redo
- **Ctrl** + **Enter**: Execute the selected line or content.
- **Ctrl** + **Alt** + **F**: Flag
- **Ctrl** + **Shift** + **K**: Search for the previous one.
- **Ctrl** + **K**: Search for the next one.
- **Ctrl** + **Backspace**: Delete the word to the left of the cursor.
- **Ctrl** + **Delete**: Delete the word to the right of the cursor.
- **Alt** + **Backspace**: Delete all content from the beginning of the line to the cursor.
- **Alt** + **Delete**: Delete all content from the cursor to the end of the line.
- **Alt** + **Shift**-**Left**: Select all content from the beginning of the line to the cursor.
- **Alt** + **Shift**-**Right**: Select all content from the cursor to the end of the line.

– System functions

To view the functions supported by this type of data connection, click **System Functions** on the right of the editor. You can double-click a function to the editor to use it.

– Script parameters

Enter script parameters in the SQL statement and click **Parameter Setup** in the right pane of the editor and then click **Update from Script**. You can also directly configure parameters and constants for the job script.

In the following script example, *str1* indicates the parameter name. It can contain only letters, digits, hyphens (-), underscores (_), greater-than signs (>), and less-than signs (<), and can contain a maximum of 16 characters. The parameter name must be unique.

```
select ${str1} from data;
```

– Visualized reading of data tables to generate SQL statements

Click **Data Tables** on the right of the editor to display all the tables in the current database or schema. You can select tables and columns and click **Generate SQL Statement** in the lower right corner to generate an SQL statement, which you need to manually format.

7. (Optional) In the upper part of the editor, click **Format** to format SQL statements.
8. In the upper part of the editor, click **Execute**. If you need to execute some SQL statements separately, select the SQL statements first. After executing the SQL statements, view the execution history and result of the script in the lower part of the editor.

   NOTE

   You can click **View Log** to view logs of the job.
9. Above the editor, click to save the job.

Configuring job parameters

Job parameters can be globally used in any node in jobs. The procedure is as follows:

Click **Parameter Setup** on the right of the editor and set the parameters described in **Table 6-33**.

Table 6-33 Job parameter setup

Module	Description
Variables	
Add	Click **Add** and enter the variable parameter name and parameter value in the text boxes. • Parameter Only letters, numbers, hyphens, and underscores (_) are allowed. • Parameter Value – The string type of parameter value is a character string, for example, **str1**. – The numeric type of parameter value is a number or operation expression. After the parameter is configured, it is referenced in the format of **$**{*parameter name*} in the job.
Edit Parameter Expression	Click next to the parameter value text box. In the displayed dialog box, edit the parameter expression. For more expressions, see **Expression Overview**.
Modifying a Job	Change the parameter name or value in the corresponding text boxes.
Mask	If the parameter value is a key, click to mask the value for security purposes.

Module	Description
Delete	Click next to the parameter name and value text boxes to delete the job parameter.
Constant Parameter	
Add	Click **Add** and enter the constant parameter name and parameter value in the text boxes. • Parameter Only letters, numbers, hyphens, and underscores (_) are allowed. • Parameter Value – The string type of parameter value is a character string, for example, **str1**. – The numeric type of parameter value is a number or operation expression. After the parameter is configured, it is referenced in the format of **$**{*parameter name*} in the job.
Edit Parameter Expression	Click  next to the parameter value text box. In the displayed dialog box, edit the parameter expression. For more expressions, see **Expression Overview**.
Modifying a Job	Modify the parameter name and parameter value in text boxes and save the modifications.
Delete	Click next to the parameter name and value text boxes to delete the job parameter.
Workspace Environment Variables	
View the variables and constants that have been configured in the workspace.	

Click the **Parameter Preview** tab and configure the parameters listed in **Table 6-34**.

Table 6-34 Job parameter preview

Module	Description
Current Time	This parameter is displayed only when **Scheduling Type** is set to **Run once**. The default value is the current time.
Event Triggering Time	This parameter is displayed only when **Scheduling Type** is set to **Event-based**. The default value is the time when an event is triggered.

Module	Description
Scheduling Period	This parameter is displayed only when **Scheduling Type** is set to **Run periodically**. The default value is the scheduling period.
Start Time	This parameter is displayed only when **Scheduling Type** is set to **Run periodically**. The value is the configured job execution time.
Start Time	This parameter is displayed only when **Scheduling Type** is set to **Run periodically**. The value is the time when the periodic job scheduling starts.
Subsequent Instances	Number of job instances scheduled. • The default value is **1** when **Scheduling Type** is set to **Run once**. • The default value is **1** when **Scheduling Type** is set to **Event-based**. • When **Scheduling Type** is set to **Run periodically**: If the number of instances exceeds 10, a maximum of 10 instances can be displayed, and the system displays message "A maximum of 10 instances are supported."

NOTE

In **Parameter Preview**, if a job parameter has a syntax error, the system displays a message.

If a parameter depends on the data generated during job execution, such data cannot be simulated and displayed in **Parameter Preview**.

Monitoring Quality

Single-task Data Migration and real-time jobs cannot be associated with quality jobs.

Two execution modes are available: parallel and serial. Click the **Quality Monitoring** tab on the right of the canvas to expand the slide-out panel and configure the parameters listed in **Table 6-35**.

Table 6-35 Quality monitoring parameters

Parameter	Description
Execution Mode	Execution mode of quality monitoring. The options are as follows: • **Parallel**: All the upstream operators of the quality job operator are set as primary operators. • **Serial**: Quality jobs are connected in series from top to bottom. The quality job on the top depends on the primary operator.

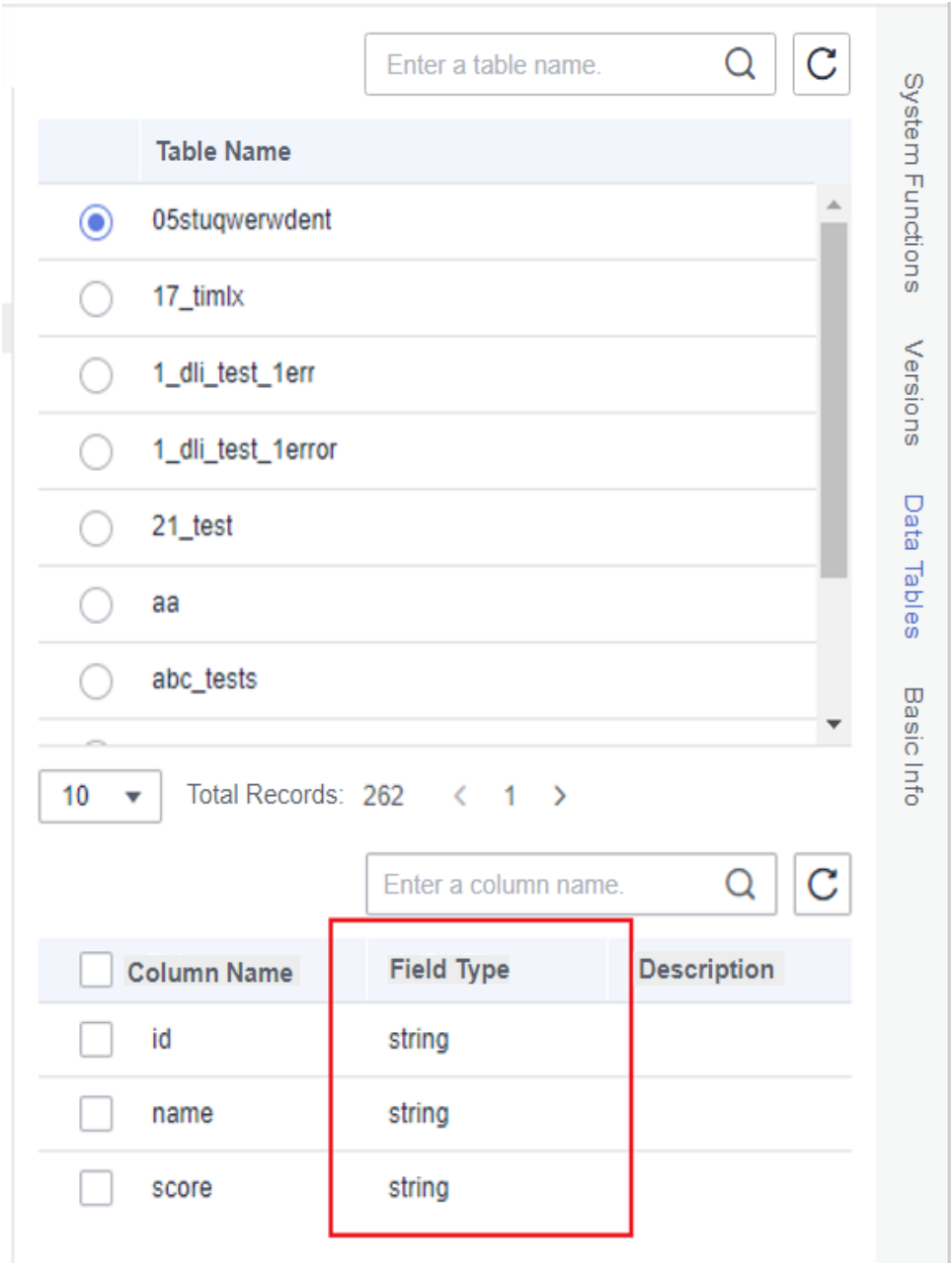
Parameter	Description
Quality job	Quality jobs to be associated with the single-task job 1. Click **Add**. The **Data Quality Monitor** slide-out panel is displayed. 2. Set a node name. 3. Set **Job Type** to **Quality job**. **NOTE** **Comparison job** is not supported. 4. Select the quality job to be associated and set other parameters based on the site requirements. If no quality job is available, create a quality job by referring to **Creating Quality Jobs**. **NOTE** • Click **Add** to add multiple quality jobs. • Click to modify an added quality job. • Click to delete an added quality job. 5. Ignore Quality Job Alarm **Yes**: Quality job alarms can be ignored. **No**: Quality job alarms cannot be ignored. When an alarm is generated, it will be reported. 6. Configure advanced settings. a. **Max. Node Execution Duration**: indicates the execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again. b. **Retry upon Failure**: specifies whether to re-execute a node if it fails to be executed. **Yes**: The node will be re-executed, and the following parameters must be configured: Retry upon Timeout **Maximum Retries** **Retry Interval (seconds)** **No**: The node will not be re-executed. This is the default value. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**. c. Policy for Handling Subsequent Nodes If the Current Node Fails

Parameter	Description
	End the current job execution plan: stops running the current job. The job instance status is **Failed**. **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. **Suspend current job execution plan**: suspends running the current job. The job instance status is **Waiting**. **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. 7. Click **OK** to complete the quality monitoring configuration.

Data Table

You can view tables of Hive SQL, Spark SQL, DLI SQL, RDS SQL, and DWS SQLsingle-task batch processing jobs. On the **Data Tables** slide-out panel, you can select a table name to view the column names, field types, and descriptions in the table.

Figure 6-34 Viewing a data table



Testing and Saving the Job

After configuring the job, perform the following operations:

Step 1 Click ▶ to execute the job.

NOTE

You can view the run logs of the job by clicking **View Log**.

Step 2 After the job is executed, click 💾 to save the job configuration.

After the job is saved, a version is automatically generated and displayed in **Versions**. The version can be rolled back. If you save a job multiple times within a

minute, only one version is recorded. If the intermediate data is important, you can click **Save new version** to save and add a version.

----End

Downloading or Dumping Script Execution Results

After a script is executed successfully, you can download or dump the execution result. By default, all users can download and dump the execution results of SQL scripts. If you do not want all users to have this permission, configure the permission by referring to **Configuring a Data Export Policy**.

- After executing a script, you can click **Download** on the **Result** tab page to download a CSV result file to a local path. You can view the download record on the **Download Center** page.
- After executing a script, you can click **Dump** on the **Result** tab page to dump a CSV and a JSON result file to OBS. For details, see **Table 6-36**.

  NOTE

  - The dump function is supported only if the OBS service is available.
  - Only the execution results of the query statements in SQL scripts can be dumped.

  Table 6-36 Dump parameters

Parameter	Mandatory	Description
Data Format	Yes	Format of the data to be exported. CSV and JSON formats are supported.
Resource Queue	No	DLI queue where the export operation is to be performed. Set this parameter when a DLI or SQL script is created.
Compression Format	No	Format of compression. Set this parameter when a DLI or SQL script is created. • none • bzip2 • deflate • gzip
Storage Path	Yes	OBS path where the result file is stored. After selecting an OBS path, customize a folder. Then, the system will create it automatically for storing the result file. You can also go to the **Download Center** page to set the default OBS path, which will be automatically set for **Storage Path** in the **Dump Result** dialog box.

Parameter	Mandatory	Description
Cover Type	No	If a folder that has the same name as your custom folder exists in the storage path, select a cover type. Set this parameter when a DLI or SQL script is created. • **Overwrite**: The existing folder will be overwritten by the customized folder. • **Report**: The system reports an error and suspends the export operation.
Export Column Name	No	**Yes**: Column names will be exported. **No**: Column names will not be exported.
Character Set	No	• **UTF-8**: default character set • **GB2312**: recommended when the data to be exported contains Chinese character sets • **GBK**: expanded based on and compatible with GB2312

Download or dump allows you to view more SQL script execution results. **Table 6-37** lists the maximum number of results that you can view, dump, and downloaded for different types of SQL scripts.

Table 6-37 Maximum number of results that you can view, dump, and download

SQL Type	Maximum Number of Results That You Can View Online	Maximum Number of Results That You Can Download	Maximum Number/Size of Results That Can Be Dumped
DLI	1,000	1,000	Unlimited
Hive	1,000	1,000	10,000 records or 3 MB
DWS	1,000	1,000	10,000 records or 3 MB
Spark	1,000	1,000	10,000 records or 3 MB
RDS	1,000	1,000	Not supported

6.4.5 Developing a Real-Time Processing Single-Task Flink SQL Job

This section describes how to develop and configure a job.

For details about how to develop a real-time processing Flink SQL job in single-task mode, see sections **Developing an SQL Script**, **Configuring Job Parameters**, **Saving a Job**, and **Templates**.

Prerequisites

- You have created a job by referring to **Creating a Job**.
- You have locked the job. Otherwise, you must click **Lock** so that you can develop the job. A job you create or import is locked by you by default. For details, see the **lock function**.

Developing an SQL Script

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. In the job directory, double-click the name of a single-task job to access the job development page.
5. On the right of the SQL editor, click **Basic Info** to configure basic information, properties, and advanced settings of the job. **Table 6-38** lists the basic information, **Table 6-39** lists the properties, and **Table 6-40** lists the advanced settings.

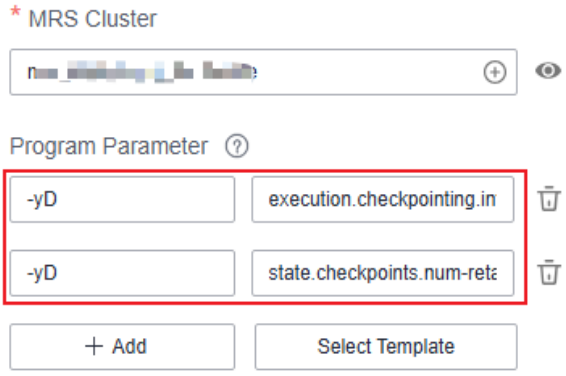
Table 6-38 Basic job information

Parameter	Description
Owner	An owner configured during job creation is automatically matched. This parameter value can be modified.
Executor	This parameter is available when **Scheduling Identities** is set to **Yes**. User that executes the job. When you enter an executor, the job is executed by the executor. If the executor is left unspecified, the job is executed by the user who submitted the job for startup.
Job Agency	This parameter is available when **Scheduling Identities** is set to **Yes**. After an agency is configured, the job interacts with other services as an agency during job execution.
Priority	Priority configured during job creation is automatically matched. This parameter value can be modified.

Parameter	Description
Execution Timeout	Timeout of the job instance. If this parameter is set to 0 or is not set, this parameter does not take effect. If the notification function is enabled for the job and the execution time of the job instance exceeds the preset value, the system sends a specified notification, and the job keeps running.
Exclude Waiting Time from Instance Timeout Duration	Whether to exclude the wait time from the instance execution timeout duration If you select this option, the time to wait before an instance starts running is excluded from the timeout duration. You can modify this setting on the **Default Configuration** page. If you do not select this option, the time to wait before an instance starts running is included in the timeout duration.
Custom Parameter	Set the name and value of the parameter.
Job Tag	Configure job tags to manage jobs by category. Click **Add** to add a tag to the job. You can also select a tag configured in **Managing Job Tags**.

Table 6-39 Properties of a single-task job

Property	Description
Flink SQL properties	
Flink Job Name	Enter the Flink job name. The name is automatically generated in *Workspace-Job name* format. **NOTE** It can contain only letters, digits, hyphens (-), and underscores. A maximum of 64 characters are allowed, and Chinese characters are not allowed.
MRS Cluster	Select an MRS cluster. **NOTE** Currently, jobs with a single Flink SQL node support MRS 3.2.0-LTS.1 and later versions.

Property	Description
Program Parameter	Set the job running parameters. This parameter is displayed only after an MRS cluster is selected. (Optional) Configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. **CAUTION** You can query historical checkpoints and select a specified checkpoint to start a real-time Flink SQL job. To make a Flink checkpoint take effect, configure the following two parameters: **Figure 6-35** Configuring program parameters  • Checkpoint interval: **-yD: execution.checkpointing.interval=1000** • Number of reserved checkpoints: **-yD: state.checkpoints.num-retained=10** When querying the checkpoint list, enter parameter **-s** and click the parameter value text box. The parameter value will be automatically displayed. **NOTE** This parameter is mandatory if the cluster version is MRS 1.8.7 or later than MRS 2.0.1. Click **Select Template** and select a parameter template. You can also select multiple templates. For details about how to create templates, see **Configuring a Template**. For details about the program parameters of MRS Spark jobs, see **Running a Flink Job** in the *MapReduce Service User Guide*.
Flink Job Parameter	Set the parameters for the Flink job. Variables required for executing the Flink job. These variables are specified by the functions in the Hive script. Multiple parameters are separated by spaces.

Property	Description
MRS Resource Queue	Select a created MRS resource queue. Select a queue you configured in the queue permissions of DataArts Security. If you set multiple resource queues for this node, the resource queue you select here has the highest priority.
Rerun Policy	• Rerun from the previous checkpoint • Rerun the job
Input Data Path	Set the input data path. You can select an HDFS or OBS path.
Output Data Path	Set the output data path. You can select an HDFS or OBS path.

Table 6-40 Advanced Settings

Parameter	Mandatory	Description
Job Status Polling Interval (s)	Yes	Set the interval at which the system checks whether the job is complete. The interval can range from 30s to 60s, or 120s, 180s, 240s, or 300s. During job execution, the system checks the job status at the configured interval.
Maximum Wait Time	Yes	Set the timeout interval for the job. If the job is not complete within the timeout interval and retry is enabled, the job will be executed again. **NOTE** If the job is in starting state and fails to start, it will fail upon timeout.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute the job if it fails • **Yes**: The job will be re-executed if it fails. Configure the following parameters: – Retry upon Timeout – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The job will not be re-executed if it fails. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.

6. Enter one or more SQL statements in the SQL editor.

   NOTE

   - SQL statements are separated by semicolons (**;**). If semicolons are used in other places but not used to separate SQL statements, escape them with backslashes (****). The following is an example:
     ```
     select 1;
     select * from a where b="dsfa\;";  --example 1\;example 2.
     ```
   - The script cannot be larger than 16 MB.
   - The system date obtained by using an SQL statement is different from that obtained by using the database tool. The query result is stored in the database in the YYYY-MM-DD format, but the query result displayed on the page is in the converted format.

   To facilitate script development, DataArts Factory provides the following capabilities:

   – The script editor supports the following shortcut keys, which improve the script development efficiency:

     - **F8**: Run a script.
     - **F9**: Stop running a script.
     - **Ctrl** + **/**: Comment out or uncomment the line or code block where the cursor resides.
     - **Ctrl** + **Z**: Undo an action.
     - **Ctrl** + **F**: Search for information.
     - **Ctrl** + **Shift** + **R**: Replace

- **Ctrl** + **X**: Cut
- **Ctrl** + **S**: Save a script.
- **Alt** + mouse dragging: Select columns to edit a block.
- **Ctrl** + mouse click: Select multiple lines to edit or indent them together.
- **Shift** + **Ctrl** + **K**: Delete the current line.
- **Ctrl** + → (or ←): Move the cursor rightwards (or leftwards) by word.
- **Ctrl** + **Home** or **Ctrl** + **End**: Navigate to the beginning or end of the current file.
- **Home** or **End**: Navigate to the beginning or end of the current line.
- **Ctrl** + **Shift** + **L**: Double-click all the same character strings and add cursors to them to implement batch modification.
- **Ctrl** + **D**: Delete a line.
- **Shift** + **Ctrl** + **U**: Unlock a script.
- **Ctrl** + **Alt** + **K**: Select the word where the cursor resides.
- **Ctrl** + **B**: Format
- **Ctrl** + **Shift** + **Z**: Redo
- **Ctrl** + **Enter**: Execute the selected line or content.
- **Ctrl** + **Alt** + **F**: Flag
- **Ctrl** + **Shift** + **K**: Search for the previous one.
- **Ctrl** + **K**: Search for the next one.
- **Ctrl** + **Backspace**: Delete the word to the left of the cursor.
- **Ctrl** + **Delete**: Delete the word to the right of the cursor.
- **Alt** + **Backspace**: Delete all content from the beginning of the line to the cursor.
- **Alt** + **Delete**: Delete all content from the cursor to the end of the line.
- **Alt** + **Shift**-**Left**: Select all content from the beginning of the line to the cursor.
- **Alt** + **Shift**-**Right**: Select all content from the cursor to the end of the line.

– Script parameters

Enter script parameters in the SQL statement and click **Parameter Setup** in the right pane of the editor and then click **Update from Script**. You can also directly configure parameters and constants for the job script.

In the following script example, *str1* indicates the parameter name. It can contain only letters, digits, hyphens (-), underscores (_), greater-than signs (>), and less-than signs (<), and can contain a maximum of 16 characters. The parameter name must be unique.

```
select ${str1} from data;
```

7. (Optional) In the upper part of the editor, click **Format** to format SQL statements.
8. Above the editor, click  to save the job and submit it.

Configuring Job Parameters

Job parameters can be globally used in any node in jobs. The procedure is as follows:

Click **Parameters** on the right of the editor and set the parameters described in **Table 6-41**.

Table 6-41 Job parameters

Function	Description
Variables	
Add	Click **Add** and enter the variable parameter name and parameter value in the text boxes. • Parameter name Only letters, digits, hyphens (-), and underscores (_) are allowed. • Parameter value – The string type of parameter value is a character string, for example, **str1**. – The numeric type of parameter value is a number or operation expression. After the parameter is configured, it is referenced in the format of **${***parameter name***}** in the job.
Edit Parameter Expression	Click  next to the parameter value text box. In the displayed dialog box, edit the parameter expression. For more expressions, see **Expression Overview**.
Modify	Change the parameter name or value in the corresponding text boxes.
Mask	If the parameter value is a key, click  to mask the value for security purposes.

Function	Description
Delete	Click next to the parameter name and value text boxes to delete the job parameter.
Constant Parameter	
Add	Click **Add** and enter the constant parameter name and parameter value in the text boxes. • Parameter name Only letters, digits, hyphens (-), and underscores (_) are allowed. • Parameter value – The string type of parameter value is a character string, for example, **str1**. – The numeric type of parameter value is a number or operation expression. After the parameter is configured, it is referenced in the format of **$**{*parameter name*} in the job.
Edit Parameter Expression	Click  next to the parameter value text box. In the displayed dialog box, edit the parameter expression. For more expressions, see **Expression Overview**.
Modify	Modify the parameter name and parameter value in text boxes and save the modifications.
Delete	Click next to the parameter name and value text boxes to delete the job parameter.
Workspace Environment Variables	
View the variables and constants that have been configured in the workspace.	

Click the **Parameter Preview** tab and configure the parameters listed in **Table 6-42**.

Table 6-42 Job parameter preview

Function	Description
Current Time	This parameter is displayed only when **Scheduling Type** is set to **Run once**. The default value is the current time.
Event Triggering Time	This parameter is displayed only when **Scheduling Type** is set to **Event-based**. The default value is the time when an event is triggered.

Function	Description
Scheduling Period	This parameter is displayed only when **Scheduling Type** is set to **Run periodically**. The default value is the scheduling period.
Start Time	This parameter is displayed only when **Scheduling Type** is set to **Run periodically**. The value is the configured job execution time.
Start Time	This parameter is displayed only when **Scheduling Type** is set to **Run periodically**. The value is the time when the periodic job scheduling starts.
Subsequent Instances	Number of job instances scheduled. • The default value is **1** when **Scheduling Type** is set to **Run once**. • The default value is **1** when **Scheduling Type** is set to **Event-based**. • When **Scheduling Type** is set to **Run periodically**: If the number of instances exceeds 10, a maximum of 10 instances can be displayed, and the system displays message "A maximum of 10 instances are supported."

NOTE

In **Parameter Preview**, if a job parameter has a syntax error, the system displays a message.

If a parameter depends on the data generated during job execution, such data cannot be simulated and displayed in **Parameter Preview**.

Saving a Job

After configuring the job, perform the following operations:

Step 1 Click ▶ to execute the job.

Step 2 After the job is executed, click 💾 to save the job configuration.

After the job is saved, a version is automatically generated and displayed in **Versions**. The version can be rolled back. If you save a job multiple times within a minute, only one version is recorded. If the intermediate data is important, you can click **Save new version** to save and add a version.

----End

Templates

When developing a real-time processing, single-task Flink SQL job, you can reference a script template. For details about how to create a template, see **Configuring a Template**. For details about how to use a script template, see **Using Script Templates and Parameter Templates**.

6.4.6 Developing a Real-Time Processing Single-Task Flink Jar Job

Prerequisites

A single-task real-time processing Flink Jar job has been created. For details, see **Creating a Job**.

Configuring the Flink Jar Job

Table 6-43 Properties

Parameter	Mandatory	Description
Flink Job Name	Yes	Enter the Flink job name. The name is automatically generated in *Workspace-Job name* format. The job name can contain 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. Chinese characters are not allowed.
MRS Cluster	Yes	Select an MRS cluster. **NOTE** Currently, jobs with a single Flink Jar node support MRS 3.2.0-LTS.1 and later versions.

Parameter	Mandatory	Description
Program Parameter	No	Set job running parameters. This parameter is displayed only after an MRS cluster is selected. (Optional) Configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. **CAUTION** You can query historical checkpoints and select a specified checkpoint to start a Flink JAR job. To make a Flink checkpoint take effect, configure the following two parameters: • Checkpoint interval: **-yD: execution.checkpointing.interval=1000** • Number of reserved checkpoints: **-yD: state.checkpoints.num-retained=10** When querying the checkpoint list, enter parameter **-s** and click the parameter value text box. The parameter value will be automatically displayed. **NOTE** This parameter is mandatory if the cluster version is MRS 1.8.7 or later than MRS 2.0.1. Click **Select Template** and select a parameter template. You can also select multiple templates. For details on how to create data connections, see **Configuring a Template**. For details about the program parameters of MRS Spark jobs, see **Running a Flink Job** in the *MapReduce Service User Guide*.
Job Execution Parameter	No	Set the parameters for the Flink job. Variables required for executing the Flink job. These variables are specified by the functions in the Hive script. Multiple parameters are separated by spaces.
MRS Resource Queue	No	Select a created MRS resource queue. Select a queue you configured in the queue permissions of DataArts Security. If you set multiple resource queues for this node, the resource queue you select here has the highest priority.
Flink job resource package	Yes	Select a JAR package. Before selecting a JAR package, upload the JAR package to the OBS bucket, create a resource on the **Manage Resource** page, and add the JAR package to the resource management list. For details, see **Creating a Resource**.
Rerun Policy	No	• Rerun from the previous checkpoint • Rerun the job

Parameter	Mandatory	Description
Input Data Path	No	Set the input data path. You can select an HDFS or OBS path.
Output Data Path	No	Set the output data path. You can select an HDFS or OBS path.

Table 6-44 Advanced settings

Parameter	Mandatory	Description
Job Status Polling Interval (s)	Yes	Set the interval at which the system checks whether the job is complete. The interval can range from 30s to 60s, or 120s, 180s, 240s, or 300s. During job execution, the system checks the job status at the configured interval.
Maximum Wait Time	Yes	Set the timeout interval for the job. If the job is not complete within the timeout interval and retry is enabled, the job will be executed again. **NOTE** If the job is in starting state and fails to start, it will fail upon timeout.
Retry upon Failure	No	Whether to re-execute a node if it fails to be executed. ● **Yes**: The node task will be re-executed, and the following parameters must be configured: – Retry upon Timeout – **Maximum Retries** – **Retry Interval (seconds)** ● **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.

6.4.7 Developing a Real-Time Processing Single-Task DLI Spark Job

Prerequisites

A single-task real-time processing DLI Spark job has been created. For details, see **Creating a Job**.

Configuring a DLI Spark job

Table 6-45 Properties

Parameter	Mandatory	Description
Job Name	Yes	Enter the DLI Spark job name. The job name can contain 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed.
DLI Queue	Yes	Select a DLI queue.
Spark Version	No	• 2.3.2 • 2.4.5 • 3.1.1
Job Type	No	Type of the Spark image used by the job. The following options are available: • Basic • AI-enhanced • Image If you select this option, select an image, and its version is automatically displayed. You can create images by following the instructions in **Image Management**.
Job Running Resource	No	• 8 vCPUs, 32 GB memory • 16 vCPUs, 64 GB memory • 32 vCPUs, 128 GB memory
Major Job Class	No	Java/Scala main class of the job
Spark program resource package	Yes	Resource package on which the Spark program depends

Parameter	Mandatory	Description
Resource Type	Yes	• OBS path • DLI program package **DLI program package**: The resource package file will not be uploaded to the DLI resource management system before the job is executed. **OBS path**: The resource package file will not be uploaded to DLI resource management system before the job is executed. The OBS path where the file is located is part of the message body for starting the job. This type is recommended.
Group	No	This parameter is required when **Resource Type** is set to **DLI program package**. A Spark program resource package is uploaded to a specified group. The main JAR package and dependency package are uploaded to the same group. • **Use Existing**: Select an existing group. • **Create New**: Create a group. The group name can contain only letters, digits, periods (.), hyphens (-), and underscores (_). • **Do not use**
Major-Class Entry Parameters	No	Press **Enter** to separate parameters.
Spark program resource package	No	Enter parameters in key=value format and separate parameters by pressing **Enter**.
Module Name	No	Select one or more module names.
Metadata Access	No	Whether metadata can be accessed To access the OBS table created by the DLI SQL job in the DLI Spark job, enable metadata access.

Table 6-46 Advanced settings

Parameter	Mandatory	Description
Job Status Polling Interval (s)	Yes	Set the interval at which the system checks whether the job is complete. The interval can range from 30s to 60s, or 120s, 180s, 240s, or 300s. During job execution, the system checks the job status at the configured interval.
Maximum Wait Time	Yes	Set the timeout interval for the job. If the job is not complete within the timeout interval and retry is enabled, the job will be executed again. **NOTE** If the job is in starting state and fails to start, it will fail upon timeout.
Retry upon Failure	No	Whether to re-execute a node if it fails to be executed. • **Yes**: The node task will be re-executed, and the following parameters must be configured: – Retry upon Timeout – Maximum Retries – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.

6.4.8 Setting Up Scheduling for a Job

This section describes how to set up scheduling for an orchestrated job.

- If the processing mode of a job is batch processing, configure scheduling types for jobs. Three scheduling types are supported: run once, run periodically, and event-based. For details, see **Setting Up Scheduling for a Job Using the Batch Processing Mode**.
- If the processing mode of a job is real-time processing, configure scheduling types for nodes. Three scheduling types are supported: run once, run periodically, and event-based. For details, see **Setting Up Scheduling for Nodes of a Job Using the Real-Time Processing Mode**.

Prerequisites

- You have performed the operations in **Developing a Pipeline Job** or **Developing a Batch Processing Single-Task SQL Job**.
- You have locked the job. Otherwise, you must click **Lock** so that you can develop the job. A job you create or import is locked by you by default. For details, see the **lock function**.

Constraints

- Set an appropriate value for this parameter. A maximum of five instances can be concurrently executed in a job. If the start time of a job instance is later than the configured job execution time, the job instances in the subsequent batch will be queued. As a result, the job execution costs a longer time than expected. For CDM and ETL jobs, the recurrence must be at least 5 minutes. In addition, the recurrence should be adjusted based on the data volume of the job table and the update frequency of the source table.
- If you use DataArts Studio DataArts Factory to schedule a CDM migration job and configure a scheduled task for the job in DataArts Migration, both configurations take effect. To ensure unified service logic and avoid scheduling conflicts, enable job scheduling in DataArts Factory and do not configure a scheduled task for the job in DataArts Migration.

Setting Up Scheduling for a Job Using the Batch Processing Mode

Three scheduling types are available: **Run once**, **Run periodically**, and **Event-based**. The procedure is as follows:

Click the **Scheduling Setup** tab on the right of the canvas to expand the configuration page and configure the scheduling parameters listed in **Table 6-47**.

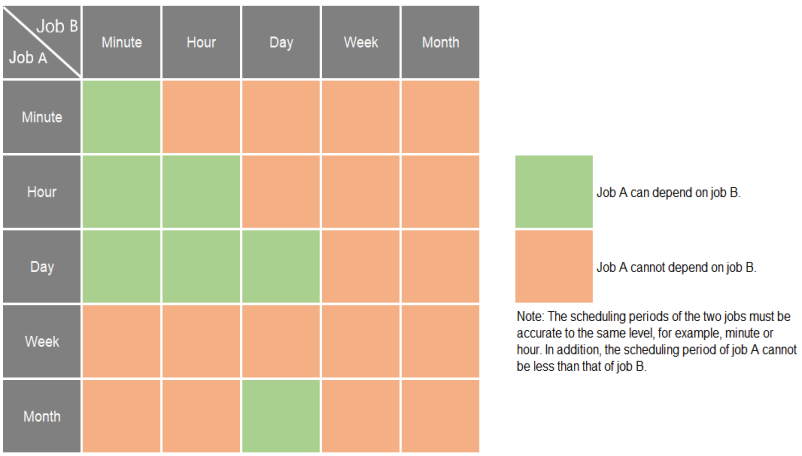
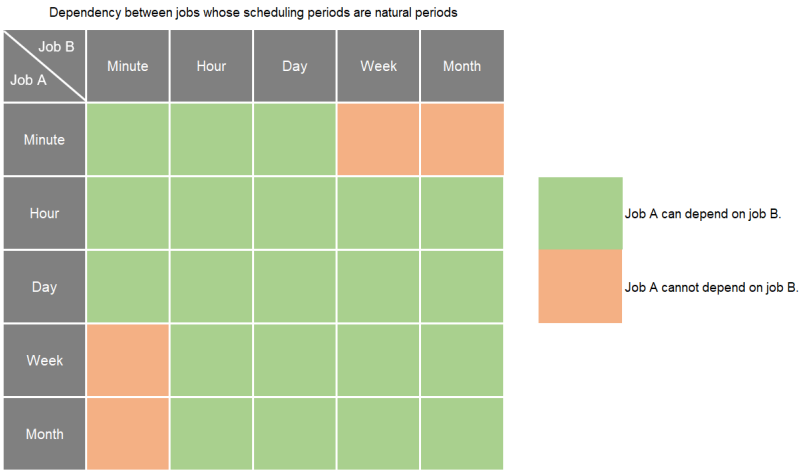
Table 6-47 Job scheduling parameters

Parameter	Description
Scheduling Type	Scheduling type of the job. Available options include: • **Run once**: You need to manually execute the job. • **Run periodically**: The job is executed periodically. For details about the parameters, see **Table 6-48**. **NOTE** For jobs that are periodically scheduled, you can select **Manual confirmation**. After that, job instances on the **Monitor Instance** page are in waiting confirmation state. When you click **Execute**, the jobs are in waiting execution state. When you **rerun instances**, they are in waiting confirmation state. When you click **Execute**, the instances are in waiting execution state. In PatchData scenarios, PatchData job instances are in waiting confirmation state on the **Monitor PatchData** page. When you click **Execute** on the **Monitor Instance** page, PatchData job instances are in waiting execution state. On the **Batch Jobs** page, job instances are in waiting confirmation state. When you click **Execute**, the jobs are in waiting execution state. During job export, if **requireManualConfirmBeforeExecute** is set to **true**, **Manual confirmation** is selected by default after job import. If **requireManualConfirmBeforeExecute** is set to **false**, **Manual confirmation** is not selected by default. • **Event-based**: The job will be executed when certain external conditions are met. For details about the parameters, see **Table 6-49**. For details, see **Scheduling Jobs Across Workspaces**.
Enable Dry Run	If you select this option, the job will not be executed, and a success message will be returned.
Task Groups	Select a configured task group. For details, see **Configuring Task Groups**. **Do not select** is selected by default. After a task group is configured, you can control the number of concurrent nodes in the current workspace in a fine-grained manner. For example, if a job contains multiple nodes or patch data, you can control the number of concurrent nodes in the current workspace. Example 1: The maximum number of concurrent tasks in the task group is set to 2, and a job has five nodes. When the job runs, only two nodes are running and the other nodes are waiting to run. Example 2: The maximum number of concurrent tasks in the task group is set to 2, and the number of concurrent periods for a PatchData job is set to 5. When the PatchData job runs, two PatchData job instances are running, and the other job instances are waiting to run. The waiting instances can be delivered normally after a period of time.

Table 6-48 Parameters for jobs that are executed periodically

Parameter	Description
From and to	The period during which a scheduling task takes effect. You can set it to today or tomorrow by clicking the time box and then **Today** or **Tomorrow**.
Recurrence	The frequency at which the scheduling task is executed, which can be: Set an appropriate value for this parameter. A maximum of five instances can be concurrently executed in a job. If the start time of a job instance is later than the configured job execution time, the job instances in the subsequent batch will be queued. As a result, the job execution costs a longer time than expected. For CDM and ETL jobs, the recurrence must be at least 5 minutes. In addition, the recurrence should be adjusted based on the data volume of the job table and the update frequency of the source table. You can modify the scheduling period of a running job. • **Minutes**: The job starts at the top of the hour. The interval is accurate to minute. After the scheduling ends at the end time of the current day, the scheduling automatically starts on the next day. • **Hours**: You can select **Interval Hour**, indicating that the job starts at a specified time point and that the interval is accurate to hour. After the scheduling ends at the end time of the current day, the scheduling automatically starts on the next day. You can also select **Discrete Hour** and specify any hour in a day to schedule the job. • **Every day**: The job starts at a specified time on a day. The scheduling period is one day. • **Every week**: You can select a specified time point of one or more days in a week. • **Every month**: You can select a specified time point of one or more days in a month. You can select the last day of each month.
Scheduling Calendar	Select a scheduling calendar. The default value is **Do not use**. For details about how to configure a scheduling calendar, see **Configuring a Scheduling Calendar**. • The job is scheduled on the custom working days in the calendar. On non-working days, a dry run occurs. Examples: periodic job scheduling and PatchData tasks. • Changes to the working days of the scheduling calendar do not take effect for the job instances that are being executed, but can take effect immediately for those that have not been generated.

Parameter	Description
OBS Listening	If you enable this function, the system automatically listens to the OBS path for new job files. If you disable this function, the system no longer listens to the OBS path. Configure the following parameters: • **OBS File**: An EL expression is supported. • **Listening Interval**: Set a value ranging from 1 to 60, in minutes. • **Timeout**: Set a value ranging from 1 to 1440, in minutes.

Parameter	Description
Dependency job	You can select jobs that are executed periodically in different workspaces as dependency jobs. The current job starts only after the dependency jobs are executed. You can click **Parse Dependency** to automatically identify job dependencies. **NOTE** For details about job dependency rules across workspaces, see **Job Dependency Rule**. Currently, DataArts Factory supports two types of job dependency policies, that is, dependency between jobs whose scheduling periods are traditional periods and dependency between jobs whose scheduling periods are natural periods. You can select either of them. The scheduling periods for new DataArts Studio instances are natural periods. **Figure 6-36** Dependency between jobs whose scheduling periods are traditional periods  **Figure 6-37** Dependency between jobs whose scheduling periods are natural periods 

Parameter	Description
	For details about the conditions for setting dependency jobs and how jobs run after dependency jobs are set, see **Dependency Policies for Periodic Scheduling**.
Policy for Current job If Dependency job Fails	Policy for processing the current job when one or more instances of its dependency job fail to be executed in its period. • Pending Waits to execute the current job, which affects the execution of subsequent jobs. You can force the dependency job to be executed successfully. • Continue Continues to execute the current job. • Cancel Cancels the current job. Its status becomes **Canceled**. For example, the recurrence of the current job is 1 hour and that of its dependency jobs is 5 minutes. • If the value of this parameter is set to **Cancel**, the current job will be canceled as long as one of the 12 instances of its dependency job fails. • If the value of this parameter is set to **Continue**, the current job will be executed after the 12 instances of its dependency job are executed. **NOTE** You can set this parameter for multiple jobs in a batch. For details, see **Configuring a Default Item**. This parameter takes effect only for new jobs.
Run After Dependency job Ends	If a job depends on other jobs, the job is executed only after its dependency job instances are executed within a specified time range. If the dependency job instances are not successfully executed, the current job is in waiting state. If you select this option, the system checks whether all job instances in the previous cycle have been executed before executing the current job.
When configuring job dependencies, you can filter dependent jobs based on whether they are being scheduled.	When configuring job dependencies, you can filter dependent jobs based on whether they are being scheduled. This prevents downstream job failures caused by upstream dependent jobs not being scheduled. • All jobs • Running jobs

Parameter	Description
Cross-Cycle Dependency	Dependency between job instances • **Independent on the previous schedule cycle**: You can set **Concurrency** to set the number of job instances that are concurrently executed. If you set it to **1**, a batch is executed only after the previous batch is executed (the execution is successful, cancelled, or failed). • **Self-dependent (The job can be rescheduled only after it is executed in the current schedule cycle. Before that, the job is in Waiting state.)** • **Skip waiting instances and run the latest instance**: Skipped job instances will be canceled and not executed. If the execution of a job instance takes a long time, multiple subsequent job instances may be skipped. However, if these job instances need to be executed, skipping them may cause service logic errors. For example, if partitioned tables are required but redundant job instances are skipped, some partitioned tables may go missing. Exercise caution when selecting this option. **NOTE** • **Skip waiting instances and run the latest instance** is only supported for jobs scheduled by minute or hour. • If the number of concurrent jobs is small and no instance has been generated, blocked instances will not be skipped. • If a job with a shorter period depends on a job with a longer period, some instances may not be skipped and still be executed.
Clear Waiting Instances	• No • Yes If this parameter is not set, expired waiting job instances will be cleared based on the workspace-level configuration by default. You can set whether to clear waiting job instances based on the site requirements.

Table 6-49 Parameters for event-based jobs

Parameter	Description
Event Type	Type of the event that triggers job running • **DIS** • **KAFKA**
Parameters for DIS event-triggered jobs	
DIS Stream	Name of the DIS stream. When a new message is sent to the specified DIS stream, DataArts Factory transfers the new message to the job to trigger the job running.

Parameter	Description
Concurrent Events	Number of jobs that can be concurrently processed. The maximum number of concurrent events is 128.
Event Detection Interval	Interval at which the system detects the DIS stream for new messages. The unit of the interval can be **Seconds** or **Minutes**.
Access Policy	Select the location where data is to be accessed: ● **Access from the last location**: For the first access, data is accessed from the most recently recorded location. For the subsequent access, data is accessed from the previously recoded location. ● **Access from a new location**: Data is accessed from the most recently recorded location each time.
Failure Policy	Select a policy to be performed after scheduling fails. ● Suspend ● Ignore the failure and proceed with the next event
Parameters for KAFKA event-triggered jobs	
Connection Name	Before selecting a data connection, ensure that a Kafka data connection has been created in the **Management Center**.
Topic	Topic of the message to be sent to the Kafka.
Concurrent Events	Number of jobs that can be concurrently processed. The maximum number of concurrent events is 128.
Event Detection Interval	Interval at which the system detects the stream for new messages. The unit of the interval can be **Seconds** or **Minutes**.
Access Policy	Select the location where data is to be accessed: ● **Access from the last location**: For the first access, data is accessed from the most recently recorded location. For the subsequent access, data is accessed from the previously recoded location. ● **Access from a new location**: Data is accessed from the most recently recorded location each time.
Failure Policy	Select a policy to be performed after scheduling fails. ● Suspend ● Ignore the failure and proceed with the next event

Setting Up Scheduling for Nodes of a Job Using the Real-Time Processing Mode

Three scheduling types are available: **Run once**, **Run periodically**, and **Event-based**. The procedure is as follows:

Select a node. On the node development page, click the **Scheduling Parameter Setup** tab. On the displayed page, configure the parameters listed in **Table 6-50**.

Table 6-50 Parameters for setting up node scheduling

Parameter	Description
Scheduling Type	Scheduling type of the job. Available options include: • **Run once**: You need to manually run the job. • **Run periodically**: The job runs automatically and periodically. • **Event-based**: The job runs when certain external conditions are met.
Parameters displayed when Scheduling Type is Run periodically	
From and to	The period during which a scheduling task takes effect. You can set it to today or tomorrow by clicking the time box and then **Today** or **Tomorrow**.
Recurrence	The frequency at which the scheduling task is executed, which can be: • Minutes • Hours You can select **Interval Hour** or **Discrete Hour**. If you select **Discrete Hour**, the job can only be scheduled by natural period. • Every day • Every week • Every month You can select the last day of each month. For CDM and ETL jobs, the recurrence must be at least 5 minutes. In addition, the recurrence should be adjusted based on the data volume of the job table and the update frequency of the source table. You can modify the scheduling period of a running job.
Cross-Cycle Dependency	Dependency between job instances • Independent on the previous schedule cycle • Self-dependent (The current job can continue to run only after the previous schedule cycle is successfully finished.) • Skip waiting instances and run the latest instance
Parameters displayed when Scheduling Type is Event-based	
Event Type	Type of the event that triggers job running
DIS Stream	Name of the DIS stream. When a new message is sent to the specified DIS stream, DataArts Factory transfers the new message to the job to trigger the job running.

Parameter	Description
Connection Name	Before selecting a data connection, ensure that a Kafka data connection has been created in the **Management Center**.
Topic	Topic of the message to be sent to the Kafka.
Consumer Group	A scalable and fault-tolerant group of consumers in Kafka. Consumers in a group share the same ID. They collaborate with each other to consume all partitions of subscribed topics. A partition in a topic can be consumed by only one consumer. **NOTE** 1. A consumer group can contain multiple consumers. 2. The group ID is a string that uniquely identifies a consumer group in a Kafka cluster. 3. Each partition of each topic subscribed to by a consumer group can be consumed by only one consumer. Consumer groups do not affect each other. If you select **DIS** or **KAFKA** for **Event Type**, the consumer group ID is automatically displayed. You can also manually change the consumer group ID.
Concurrent Events	Number of jobs that can be concurrently processed. The maximum number of concurrent events is 10.
Event Detection Interval	Interval at which the system detects the DIS stream for new messages. The unit of the interval can be **Seconds** or **Minutes**.
Failure Policy	Select a policy to be performed after scheduling fails. ● Suspend ● Ignore failure and proceed

6.4.9 Submitting a Version

Submitting a version depends on the version management function of DataArts Factory.

Version management traces script and job changes, and supports version comparison and rollback. The system retains 100 latest version records. In addition, version management can be used to distinguish the development state and production state.

- Development state: Scripts or jobs have not been submitted and are used for debugging. In the development state, you can edit, save, and run scripts or jobs without affecting those being scheduled. In addition, when a job is being associated with a script or job dependency is being configured, the associated script or job will read the configuration in the development state.
- Production state: Script or jobs have been submitted and are used for formal scheduling. In formal scheduling, the latest submitted versions of scripts or jobs will be used in scenarios such as script invocation, instance rerunning, and job dependency and patch data configuration.

Prerequisites

A job has been developed.

Submitting a Job Version

If you submit a version, the latest job in the development state will be saved and submitted and overwrite the previous job version.

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

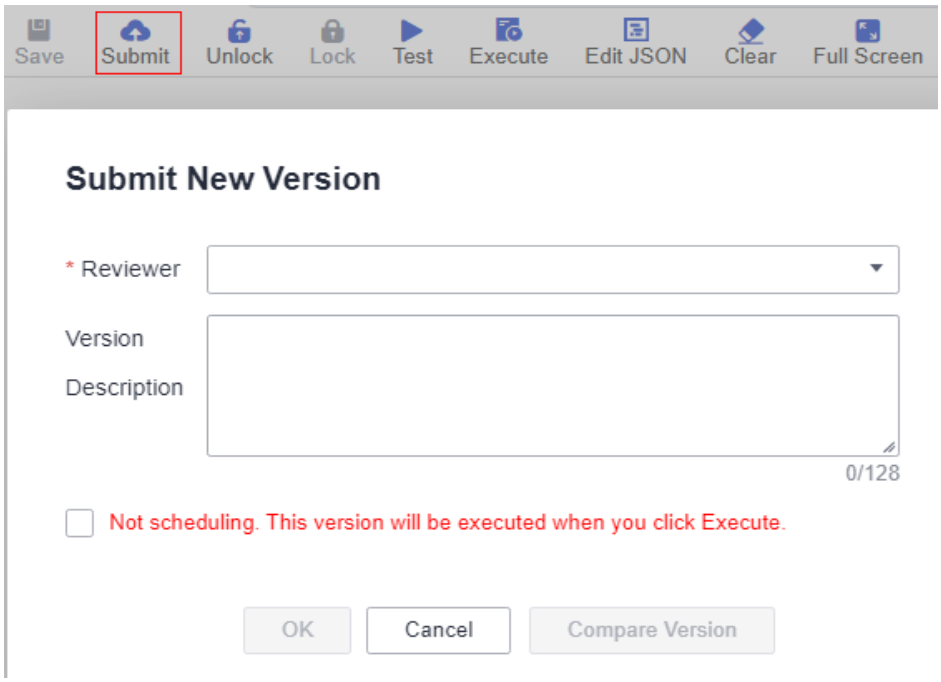
Step 2 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 3 In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.

Step 4 In the job directory, double-click the developed job to access the job development page.

Step 5 Above the job canvas or editor, click **Submit** to submit a version. In the displayed dialog box, select the reviewer, enter the change description (a maximum of 128 characters allowed), and select the check box below. If you do not select this option, you cannot click **OK**. When submitting a version, you can click **Compare Version** to view the differences between the current version and the last version.

Figure 6-38 Submitting a version



NOTE

- If review is enabled on the **Review Center** page, your submitted version will be reviewed by the workspace admin on the **Pending Review** tab page on the **Review Center** page. The version is submitted successfully only after it is approved by the admin. For details, see **Approval Settings**.

  To revoke a submitted request, go to the **Review Center** page and click the **My Applications** tab. Then you can submit an application again.
- If review is enabled, the following operations need to be reviewed: submitting jobs, deleting jobs, and importing submitted jobs.
- Before disabling the review function, ensure that there are no requests pending review in the current workspace.
- The enterprise mode does not support the review function.

----End

Version Rollback

After submitting the version, you can view it in the version list. (Currently, a maximum of 100 latest versions are saved.) Click **Roll Back** to roll back to any submitted version.

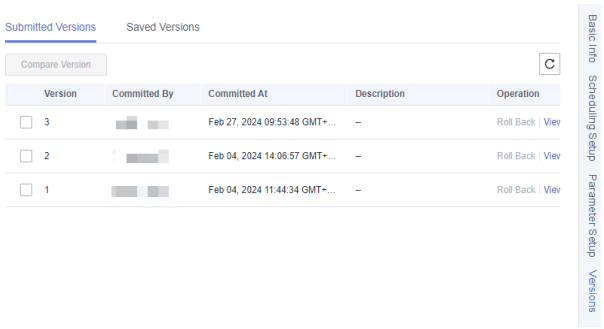
The rollback involves the following contents:

- Job definition (such as operator properties and connection lines)
- Basic job information, job scheduling configuration, job parameters, and lineage

The procedure is as follows:

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. In the job directory, double-click a job to access the job development page.
5. On the right of the page, click the **Versions** tab and view the version submission records. Select the version to be rolled back and click **Roll Back**.

Figure 6-39 Rolling back the version



Viewing Version Details

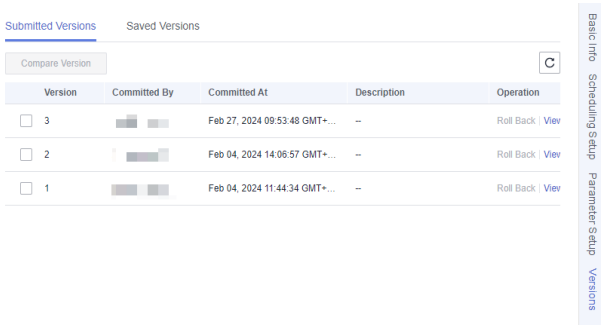
You can view the submitted version information in the version list.

The procedure is as follows:

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. In the job directory, double-click a job to access the job development page.
5. On the right of the page, click the **Versions** tab and view the version submission records. Select the desired version and click **View** to view its details.

   A new page is displayed, showing the job definition of the version. You cannot modify any job attributes in this window.

Figure 6-40 Viewing version details

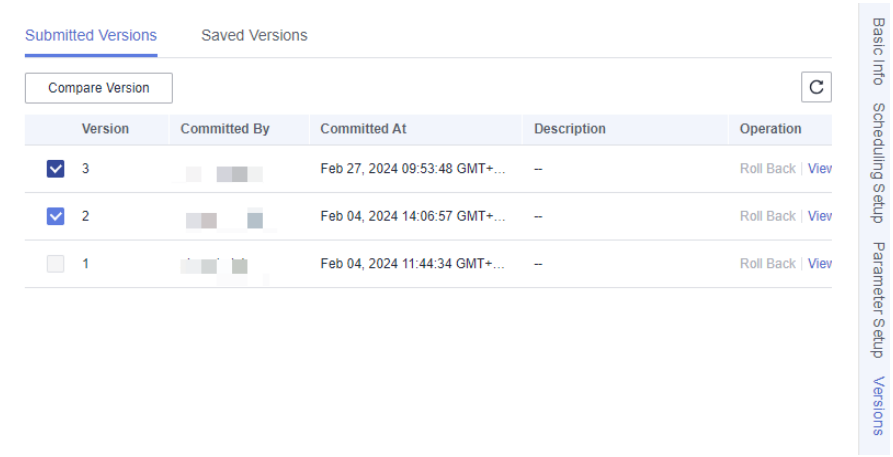


Version Comparison

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. In the job directory, double-click a job to access the job development page.
5. On the right of the page, click the **Versions** tab and view the version submission records. Select the versions to be compared and click **Compare Version**.

   If you select only one version, the selected version is compared with the JSON of the development-state job. If you select two versions, the JSON of the two versions is compared.

Figure 6-41 Comparing versions



6.4.10 Releasing a Job Task

In enterprise mode, when a developer submits a job version, the system generates a job release task. After the developer confirms the release task and the admin, deployer, a user with the DAYU Administrator or Tenant Administrator permission approves the package release request, the modified job is synchronized to the production environment.

NOTICE

- When the admin selects **Submitted** for **Job Status** during job import, a release task is generated.
- When the admin imports jobs in released state, no release task is generated.
- When a developer creates a real-time single-task job, a release task is generated for the job, and no release task is generated for the subjobs of the job.

Prerequisites

You have submitted a version. For details, see **Submitting a Version**.

Procedure

Step 1 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 2 In the left navigation pane, choose **Data Development** > **Task Release**.

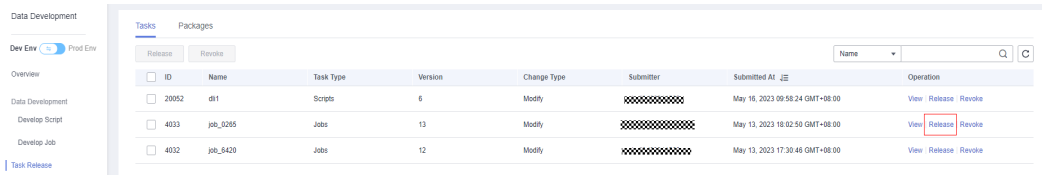
Step 3 On the **Tasks** page, the tasks generated for version submission are displayed. You can click **View** in the **Operation** column to view the modifications of a script compared with its previous version. After confirming that the modifications are correct, click **Release** to release the task.

You can filter release tasks by name or submitter. and perform fuzzy search using a task name.

NOTE

- If you have only the developer permission, the script will be synchronized to the production environment only when the task is approved by the admin or deployer.
- After clicking **Release**, set the reviewer. The reviewer must be a workspace admin, deployer, or a user with the DAYU Administrator or Tenant Administrator permission. Set at least one reviewer and do not set yourself as the reviewer. Click **Reviewer Management** to go to the **WorkSpaces** page. Click **Edit** to configure reviewers.
- You can release a maximum of 100 tasks at a time. The tasks are released asynchronously. You can view the task release process.
- After you click **Release**, the following message is displayed: "Execute jobs in the package immediately after it is released."
- You can revoke tasks not to be released as a developer, deployer, or admin.

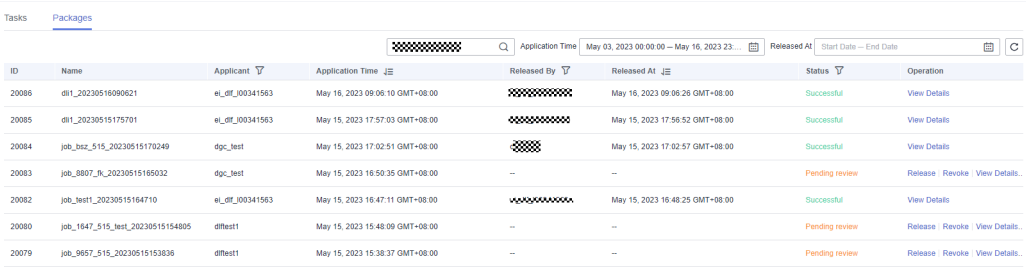
Figure 6-42 Clicking Release



Step 4 After the task is released, you can view the release status of the task on the **Packages** tab page. After approved, the task is released successfully.

You can filter release tasks by **Applicant**, **Application Time**, **Release At**, or **Released By**, and perform fuzzy search using a package name.

Figure 6-43 Viewing the task status

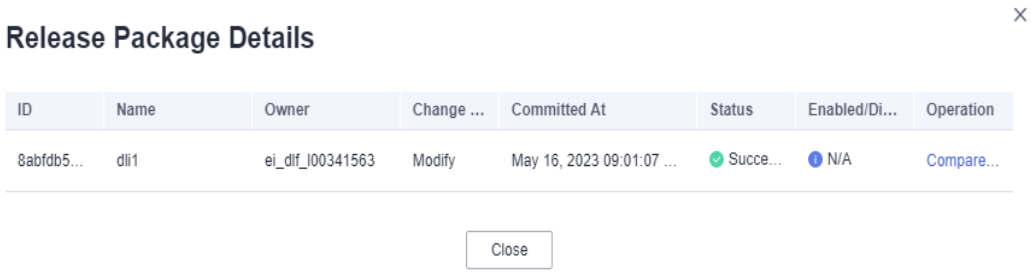


NOTE

You can revoke tasks not to be released as a developer, deployer, or admin.

After the task is released, you can click **View Details** in the **Operation** column to view the release status and startup status of the task. You can also click **Compare Version** in the **Operation** column to view the differences between different versions of release packages.

Figure 6-44 Viewing release package details



----End

6.4.11 (Optional) Managing Jobs

6.4.11.1 Copying a Job

This section describes how to copy a job.

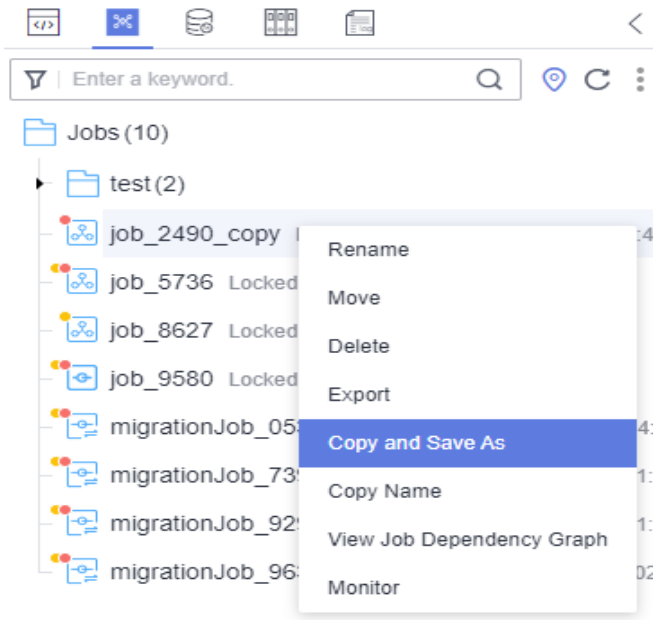
Prerequisites

A job has been developed. For details about how to develop a job, see **Developing a Pipeline Job**.

Procedure

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. In the job directory, select the job to be copied, right-click the job name, and choose **Copy Save As**.

Figure 6-45 Copying a job



5. In the displayed dialog box, configure related parameters. **Table 6-51** describes the parameters.

Table 6-51 Job and directory parameters

Parameter	Description
Job Name	Name of the job. Must consist of 1 to 128 characters and contain only letters, digits, hyphens (-), underscores (_), and periods (.).
Select Directory	Parent directory of the job directory. The parent directory is the root directory by default.

6. Click **OK**.

6.4.11.2 Copying the Job Name and Renaming a Job

You can copy the name of a job and rename a job.

Prerequisites

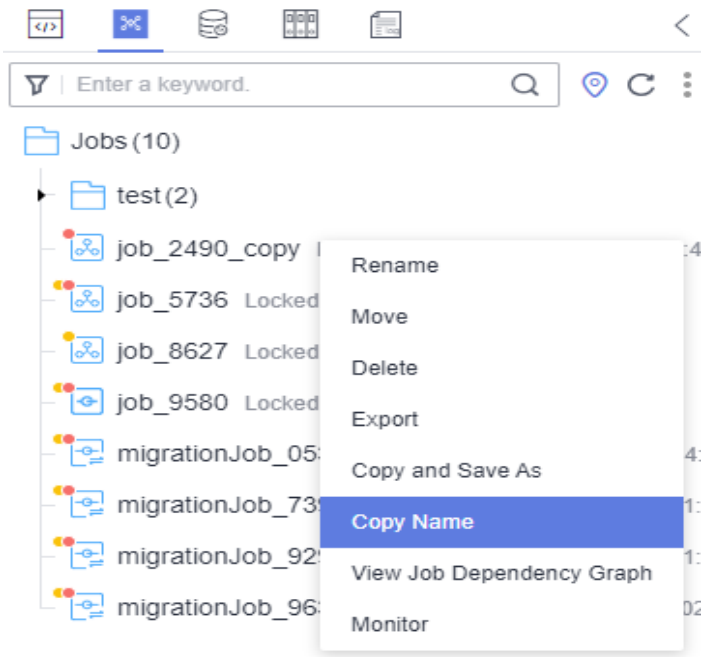
A job has been developed. For details about how to develop a job, see **Developing a Pipeline Job**.

Copying the Job Name

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. Locate the target job in the job directory, right-click the job name, and select **Copy Name** to copy the job name to the clipboard.

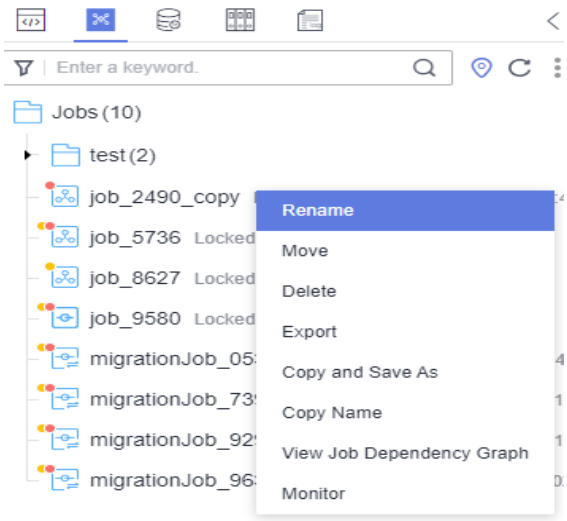
Figure 6-46 Copying the job name



Renaming a job

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. In the job directory, select the job to be renamed. Right-click the job name and choose **Rename** from the shortcut menu.

Figure 6-47 Renaming a job



5. In the displayed **Modify Job Name** dialog box, change the job name.

Figure 6-48 Renaming a job

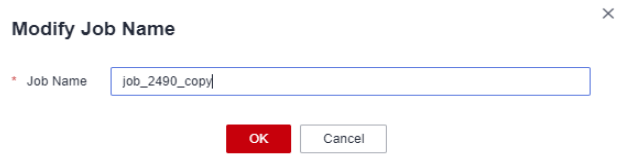


Table 6-52 Job renaming parameters

Parameter	Description
Job Name	Name of the job. Must consist of 1 to 128 characters and contain only letters, digits, hyphens (-), underscores (_), and periods (.).

6. Click **OK**.

6.4.11.3 Moving a Job or Job Directory

You can move a job file from one directory to another or move a job directory to another directory.

Prerequisites

A job has been developed. For details about how to develop a job, see **Developing a Pipeline Job**.

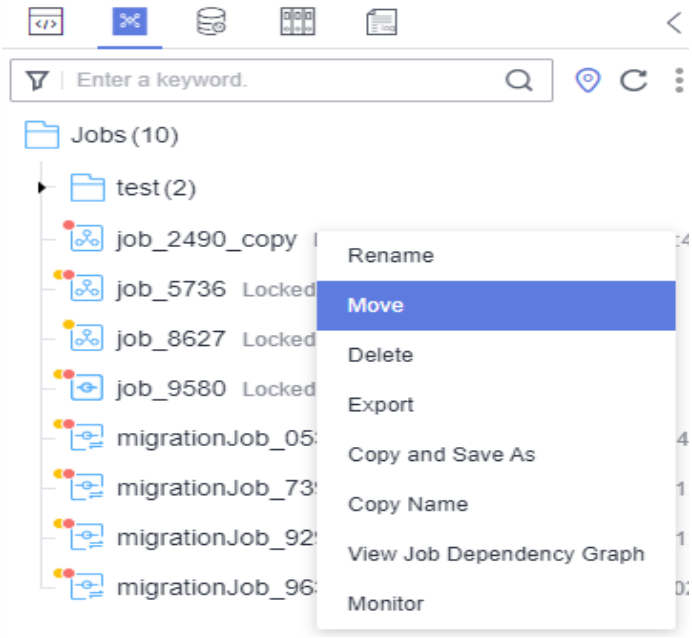
Procedure

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. Move a job or job directory.

   Method 1: right-click

   a. In the job directory, right-click a job or job folder and select **Move**.

   Figure 6-49 Selecting a job to be moved

   b. In the displayed dialog box, configure the target directory.

Figure 6-50 Moving a job

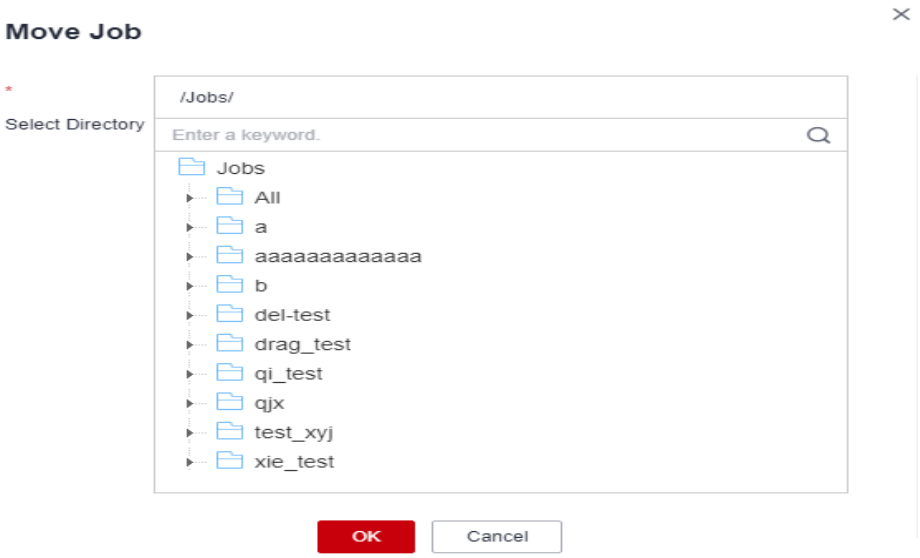


Figure 6-51 Move a directory

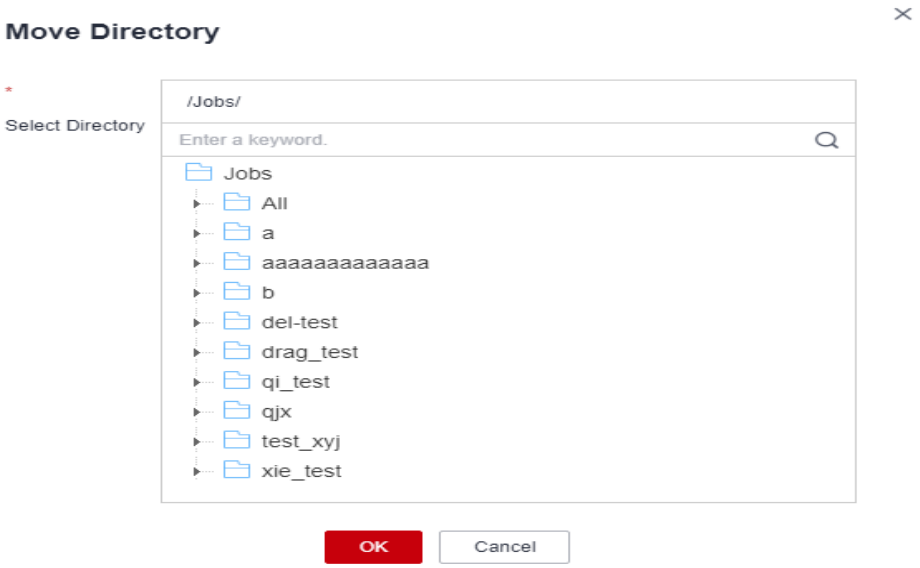


Table 6-53 Parameters for moving a job or job directory

Parameter	Description
Select Directory	Directory to which the job or job directory is to be moved. The parent directory is the **root** directory by default.

c. Click **OK**.

Method 2: drag-and-drop

Select a job or job folder and drag and drop it to the target folder.

6.4.11.4 Exporting and Importing Jobs

- Exporting jobs is to export the latest saved content in the development state.
- After a job is imported, the content in the development state is overwritten and a new version is automatically submitted.

Exporting Jobs

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

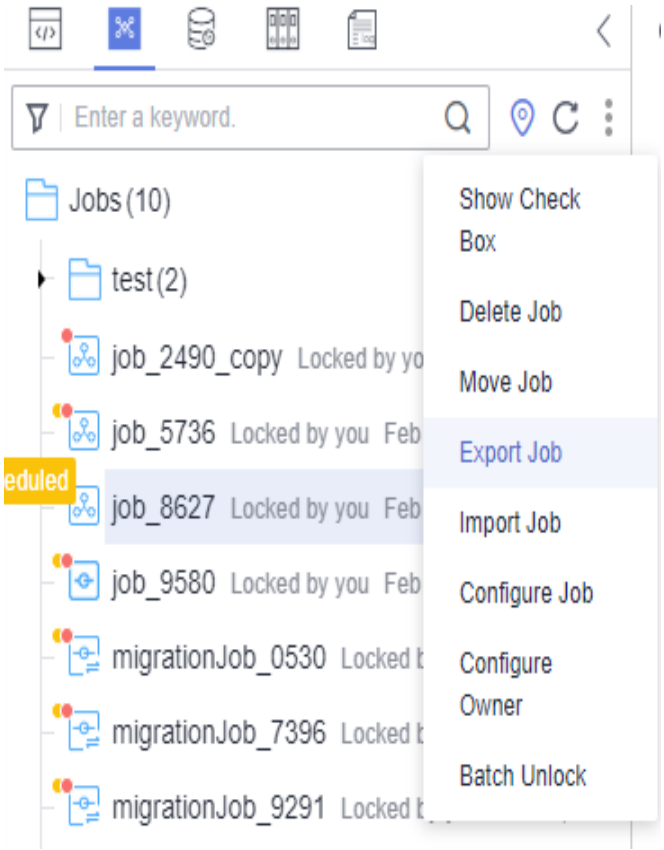
Step 2 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 3 In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.

Step 4 Click in the job directory and select **Show Check Box**.

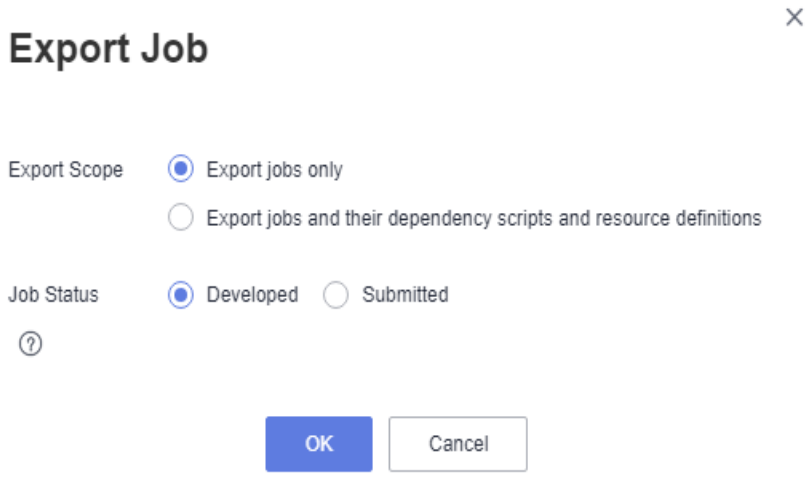
Step 5 Select jobs, click , and select Export Job. In the displayed dialog box, select **Export jobs only** or **Export jobs and their dependency scripts and resource definitions**. After the export is successful, you can obtain the exported .zip file.

Figure 6-52 Selecting and exporting jobs



Step 6 In the displayed **Export Job** dialog box, set **Export Scope** and **Job Status** and click **OK**. You can view the result in the download center.

Figure 6-53 Exporting jobs



----End

Importing Jobs

This function is available only if the OBS service is available. If OBS is unavailable, jobs can be imported from the local PC.

NOTE

- The maximum size of a job file imported from OBS is 10 MB. The maximum size of a job file imported from a local PC is 1 MB.
- If the name of a job to be imported already exists in the system, ensure that the job is in the stopped state. Otherwise, the import fails.

Import one or more jobs from the job directory.

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

Step 2 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

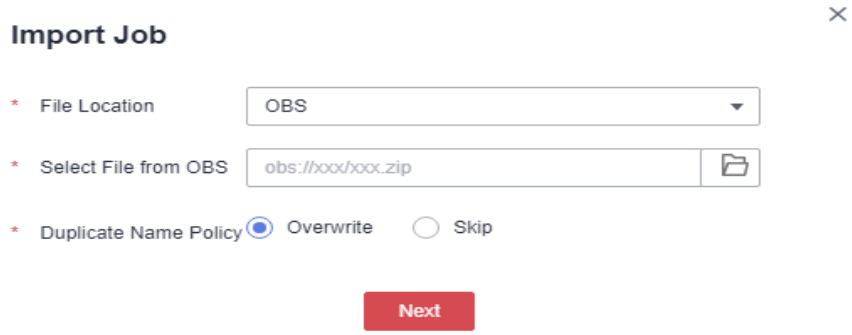
Step 3 In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.

Step 4 Click > **Import Job** in the job directory, select the job file that has been uploaded to OBS or local directory, and rename the policy.

NOTE

If you select **Overwrite** for **Duplicate Name Policy** but the hard lock policy is used and the script is locked by another user, the overwriting will fail. For details about soft and hard lock policies, see **Configuring the Hard and Soft Lock Policy**.

Figure 6-54 Importing jobs and their dependencies



Step 5 Click **Next** to import the job as instructed.

NOTE

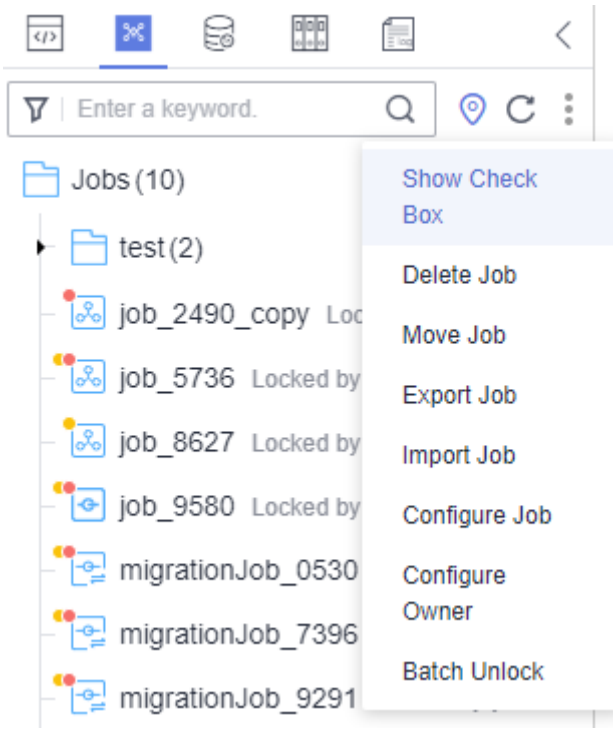
- If a job contains a tag in the locked state, the job fails to be imported.
- When a job fails to be imported and a tag needs to be automatically generated, if the tag already exists and is locked, it will not be added to the job.
- During the import, if the data connection, DIS stream, DLI queue, or GES graph associated with the job does not exist in DataArts Factory, the system prompts you to select one again.

----End

6.4.11.5 Configuring Jobs

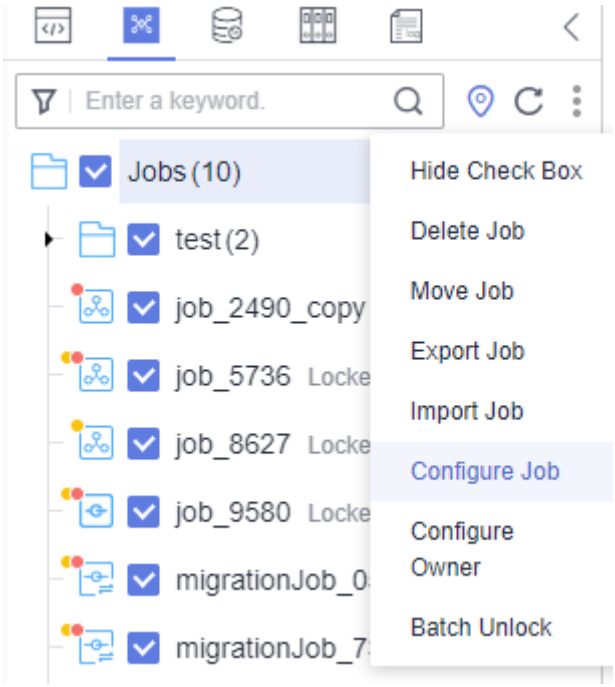
1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. Click in the job directory and select **Show Check Box**.

Figure 6-55 Clicking Show Check Box



5. Select jobs, click , and select **Configure Job**.

Figure 6-56 Configure Job



6. Configure general parameters for the jobs.

Figure 6-57 General Configuration

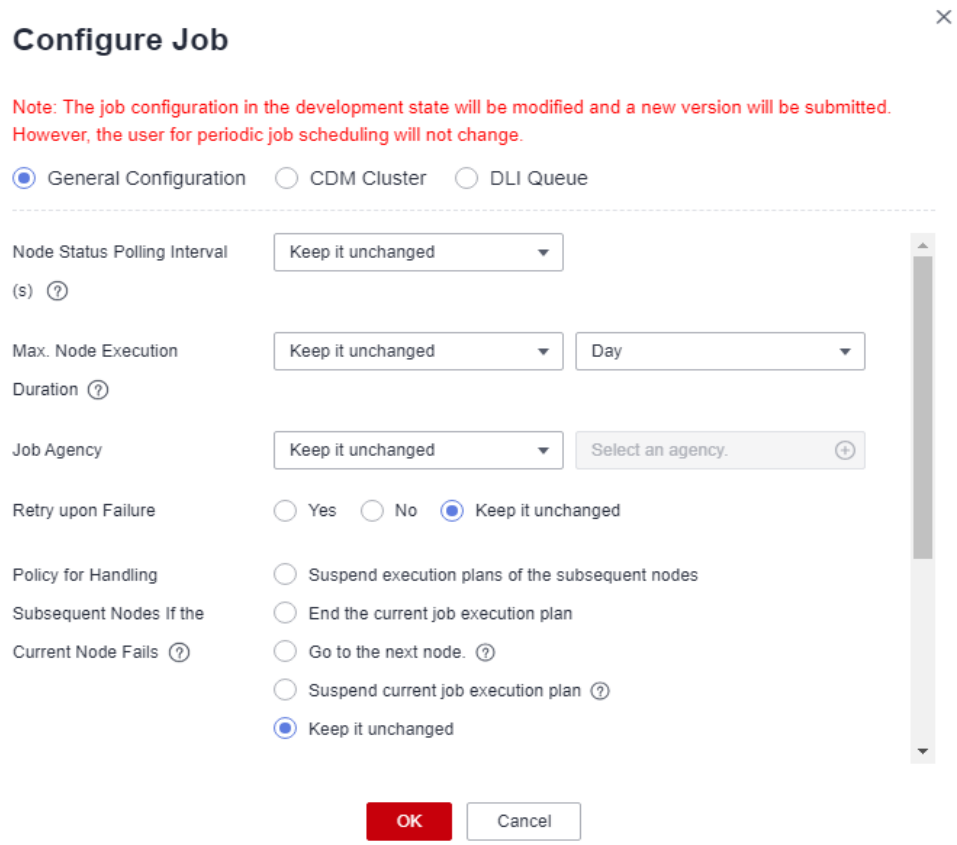


Table 6-54 General Configuration

Parameter	Description
Node Status Polling Interval	How often the system checks whether all the nodes are executed. The value ranges from 1 to 60 seconds. If you select **Keep it unchanged**, the poll interval remains unchanged for the nodes.
Max. Node Execution Duration	Maximum duration of executing the nodes of a job. When **Retry upon Failure** is set to **Yes** for a node, the node will be re-executed upon an execution failure. If you select **Keep it unchanged**, the poll interval remains unchanged for the nodes.
Job Agency	During execution of the jobs, the agency is used to communicate with other services. If you select **Keep it unchanged**, the agency remains unchanged for the jobs.

Parameter	Description
Retry upon Failure	Whether to re-execute the nodes of the selected jobs if the nodes fail to be executed. If you select **Keep it unchanged**, the retry policy remains unchanged for the nodes.
Retry upon Timeout	This parameter is displayed only when **Retry upon Failure** is set to **Yes**. Whether to re-execute the nodes of the selected jobs if the nodes time out. If you select **Keep it unchanged**, the retry policy remains unchanged for the nodes.
Policy for Handling Subsequent Nodes If the Current Node Fails	Operation to be performed if all nodes of the selected jobs fail to be executed. If you select **Keep it unchanged**, the failure policy remains unchanged for the nodes.
Action After Dependency Job Failure	Action to be taken if the dependency jobs of the selected jobs fail. This parameter is invalid if no dependency jobs have been configured for the selected jobs. If you select **Keep it unchanged**, the failure policy remains unchanged for the selected jobs.
Owner	Owner of the selected jobs, which can only be a member of the current workspace. If you select **Keep it unchanged**, the own remains unchanged for the jobs.
Concurrent Periodic Job Instances	Number of jobs that can be handled concurrently If you select **Keep it unchanged**, the number of concurrent periodic job instances remains unchanged.
Clear Waiting Instances	If you select **Yes**, you need to set **Retention Days**. If the waiting time before a job instance starts running exceeds the configured **Retention Days**, the job instance will be canceled and cleared. If you select **No**, waiting job instances will not be cleared. If you select **Keep it unchanged**, the original timeout duration rule for job instances is retained.

7. Select **CDM Cluster** and configure the CDM cluster for the CDM Job node of the selected jobs.

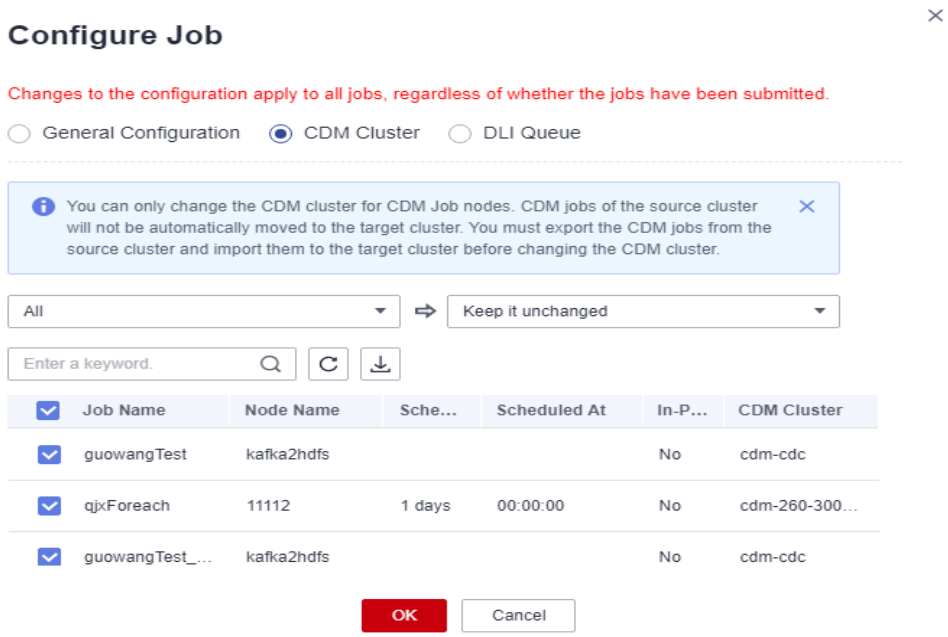
   Select the current CDM cluster from the drop-down list box on the left, and select the target CDM cluster from the drop-down list box on the right.

NOTE

1. Before migrating a CDM cluster, you must create a job with the same name in the new cluster.
2. Configure two CDM clusters for a CDM job.
   - If you select one of the source clusters, only the selected cluster will be migrated.
   - If you select both source clusters, they will be both migrated to the destination cluster.

- Search: Enter a job name and click to filter out the jobs that contain the CDM Job node.
- Refresh: Click to refresh the list of jobs that contain the CDM Job node.
- Download: Click to download the selected jobs.

Figure 6-58 CDM Cluster



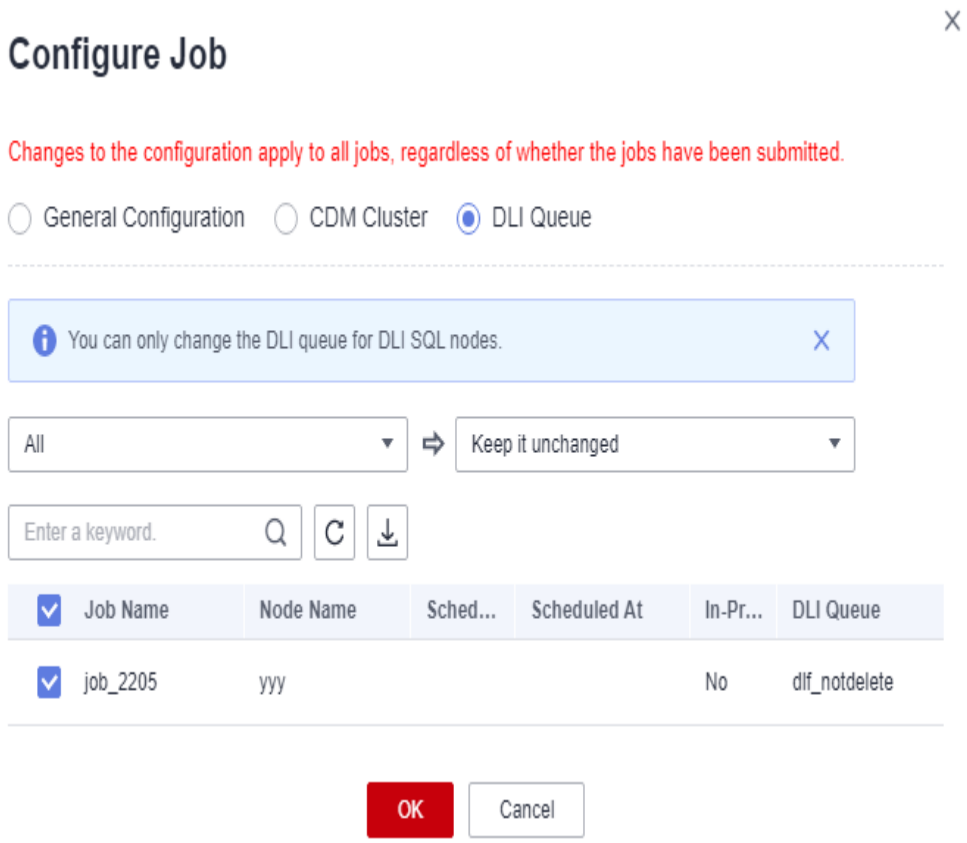
8. Select **DLI Queue** and configure the DLI queue of the DLI SQL node of the selected jobs.

   Select the current DLI queue from the drop-down list box on the left, and select the target DLI queue from the drop-down list box on the right.

   NOTE

   - Search: Enter a job name and click to filter out the jobs that contain the DLI SQL node.
   - Refresh: Click to refresh the list of jobs that contain the DLI SQL node.
   - Download: Click to download the selected jobs.

Figure 6-59 DLI Queue



9. Click **OK**.

6.4.11.6 Deleting a Job

If you do not need to use a job any more, perform the following operations to delete it to reduce the quota usage of the job.

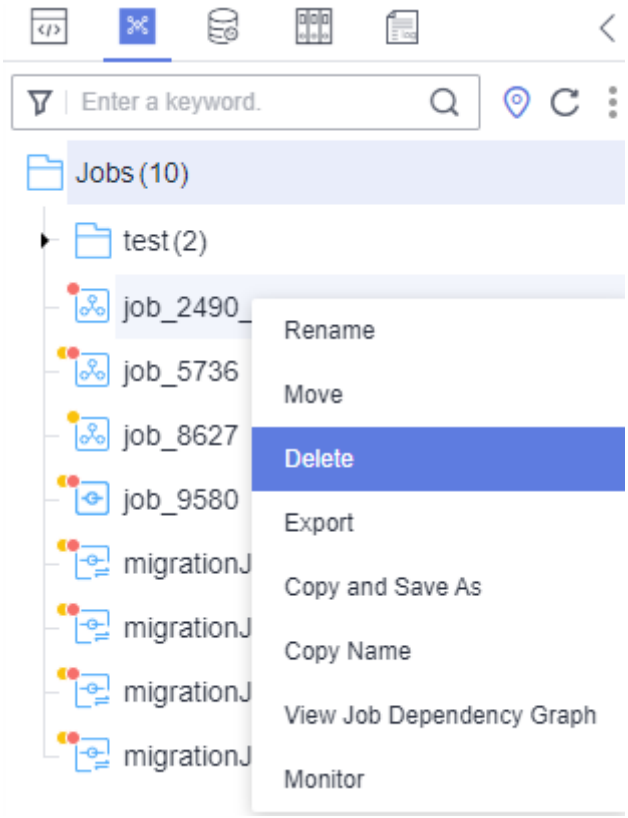
NOTE

Deleted jobs cannot be recovered. Exercise caution when performing this operation.

Deleting a Script

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. In the job directory, right-click the job that you want to delete and choose **Delete** from the shortcut menu.

Figure 6-60 Deleting a job

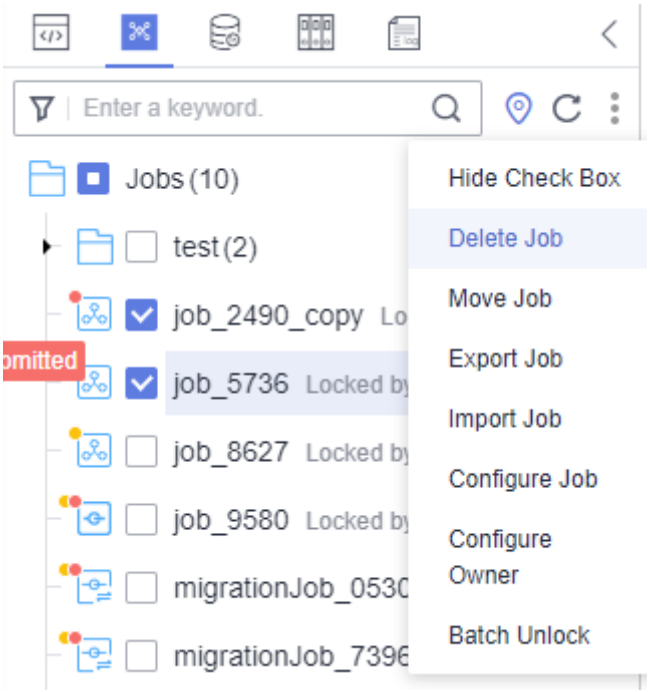


5. In the displayed dialog box, click **OK**.

Batch Deleting Scripts

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. On the top of the job directory, click and select **Show Check Box**.
5. Select the jobs to be deleted, click , and select **Batch Delete**.

Figure 6-61 Deleting jobs



6. In the displayed dialog box, click **OK**.

6.4.11.7 Unlocking a Job

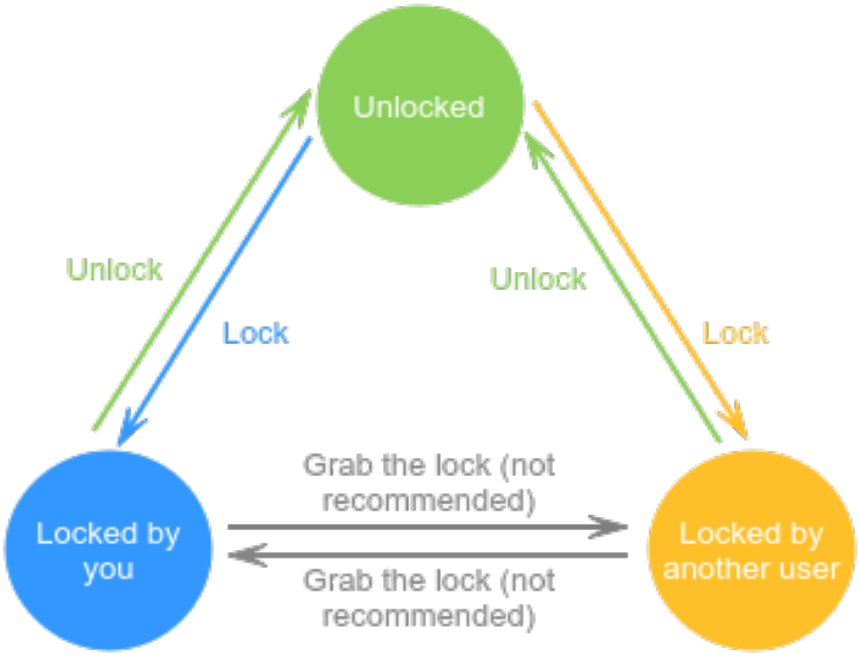
Script and job unlocking depends on the lock function of DataArts Factory.

The lock function prevents conflict caused by collaborative script or job development. If you create or import a script or job, it is locked by you by default. You can only edit, save, or submit a script or job you have locked. To edit, save, or submit a script or job that is locked by another user or not locked by any user, you must lock the script or job first.

NOTICE

- You can view the lock status of a script or job in the script or job directory tree.
- To view the latest version of an opened script or job locked by another user, you need to re-open the script or job because it is not updated in real time.
- Scripts or jobs that were created before the lock function was available are now unlocked by default. To edit, save, or submit these scripts or jobs, you need to lock them first.
- The locking operation depends on the soft and hard lock policies. For details about how to configure soft and hard lock policies, see **Configuring a Default Item**.
  - **Soft lock**: You can lock or unlock jobs or scripts, regardless of whether they are locked by others.
  - **Hard Lock**: You can lock jobs or scripts only after they have been unlocked by other users. The space administrator and the Administrator can lock and unlock jobs or scripts without any limitations.
- Do not lock a script or job that is locked by another user because if you do so, changes to the script or job made by the user will be lost. If you want to modify the script or job, contact the user to unlock the script or job, and lock it by yourself.

Figure 6-62 Lock statuses



Prerequisites

A job has been developed.

Procedure

If you submit a version, the latest job in the development state will be saved and submitted and overwrite the previous job version. You are advised to unlock the job after submitting the version so that other developers can modify the job as needed.

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

Step 2 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 3 In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.

Step 4 In the job directory, double-click the developed job to access the job development page.

Step 5 Above the job canvas or editor, click **Unlock** to unlock the job.

Figure 6-63 Unlocking a job

----End

6.4.11.8 Viewing a Job Dependency Graph

You can view a job dependency graph to learn the upstream and downstream jobs associated with the job.

Prerequisites

Dependent jobs have been configured in the job scheduling configuration in **Developing a Pipeline Job**. Otherwise, only the current job node can be displayed in the view, and the associated upstream and downstream job nodes cannot be displayed.

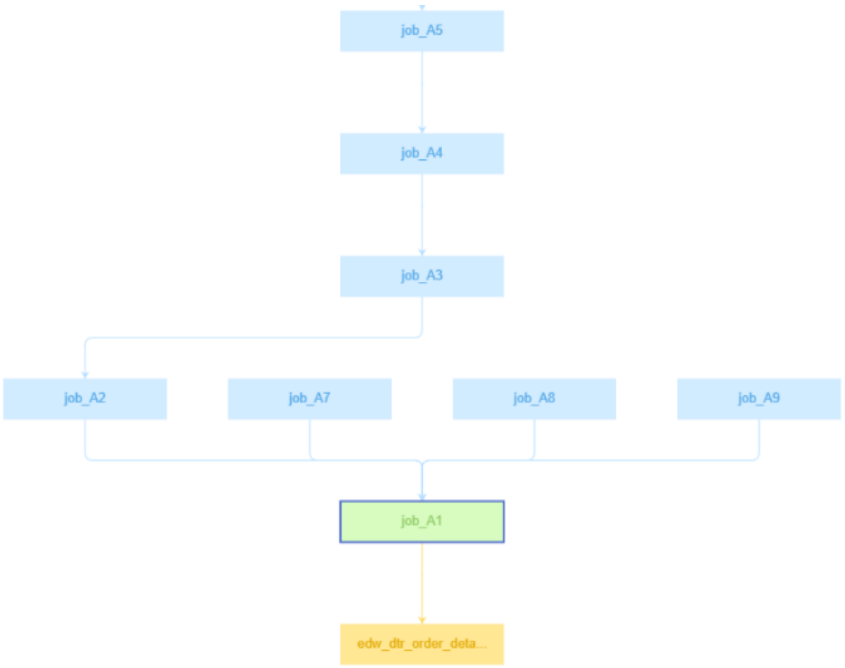
Procedure

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. In the job directory, right-click the job you want to view and choose **View Job Dependency Graph** from the shortcut menu.

Figure 6-64 Job Dependency page

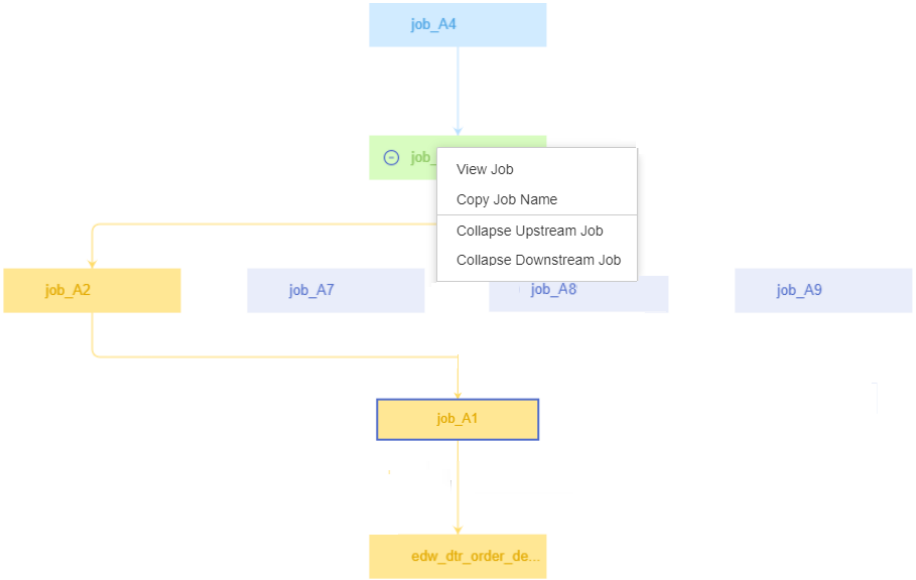
5. On the displayed **Job Dependency** page, perform any of the following operations:
   - In the upper right corner, select **Display complete dependency graphs**, **Display the current job and its upstream and downstream jobs**, or **Display the current job and its directly connected jobs**.
   - In the search box in the upper right corner, you can enter the name of a node to search for the node. The node found will be highlighted.
   - Click **Download** to download the job dependency file.
   - Scroll your mouse wheel to zoom in or zoom out the dependency graph.
   - Drag the blank area to view the complete relationship graph.
   - When the cursor is hovered on a job node, the node is marked green, its upstream job is marked blue, and its downstream job is marked yellow.

Figure 6-65 Marking upstream and downstream job nodes of a node



– Right-click a job node to view the job, copy the job name, and collapse upstream or downstream jobs.

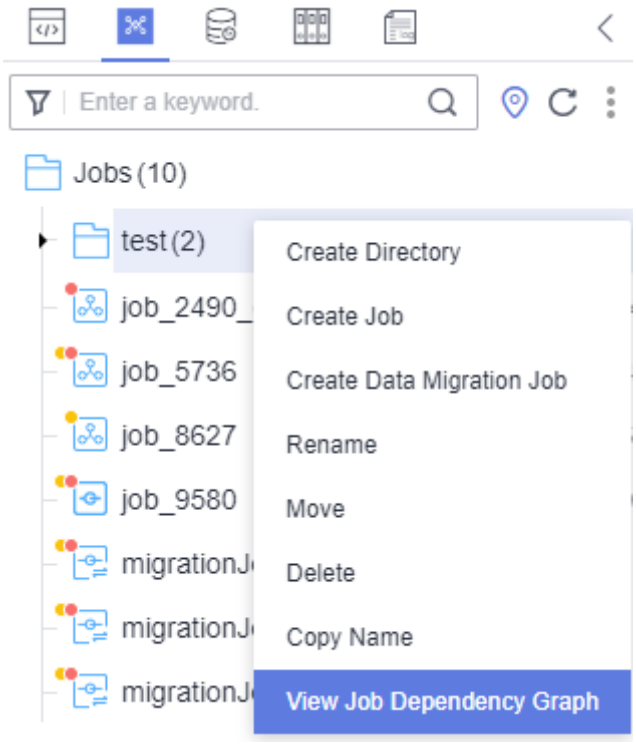
Figure 6-66 Job node operations



Viewing a Job Dependency Graph

1. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
2. Right-click the job directory and select **View Job Dependency Graph**.

Figure 6-67 Viewing the job dependency graph



3. The system displays the dependencies between all the jobs in the directory. You can search for jobs by name. The matched jobs will be highlighted.

   NOTE

   - If you click a node in the dependency graph, its upstream jobs are marked blue and its downstream jobs are marked yellow.
   - You can drag to view the full dependency graph.
   - Scroll the mouse wheel to zoom in or out the dependency graph.

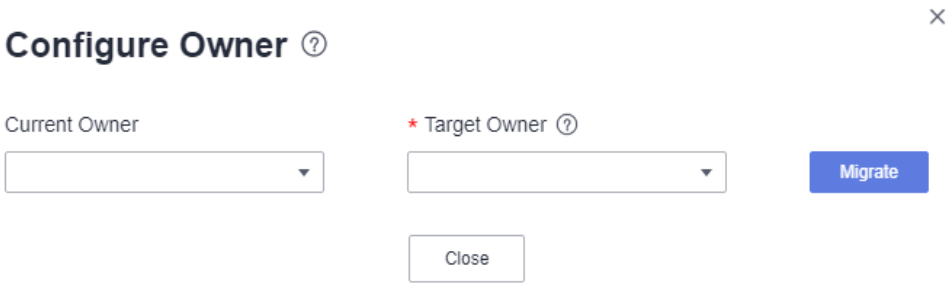
6.4.11.9 Changing the Job Owner

DataArts Factory allows you to change the job owner with a few clicks.

Procedure

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. At the top of the job directory, click and select **Configure Owner**.

Figure 6-68 Changing the owner

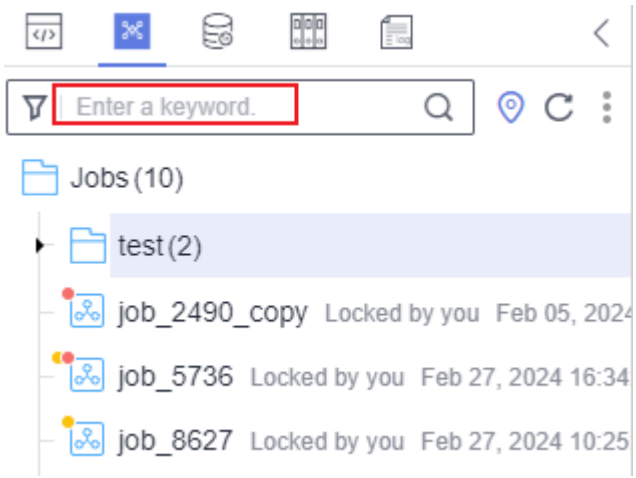


5. Set **Current Owner** and **Target Owner** and click **Migrate**.
6. When the owner is changed, click **Close**.

Related Operations

You can use an owner to filter jobs by entering the owner in the search box above the job directory.

Figure 6-69 Filtering jobs by owner



6.4.11.10 Unlocking Jobs

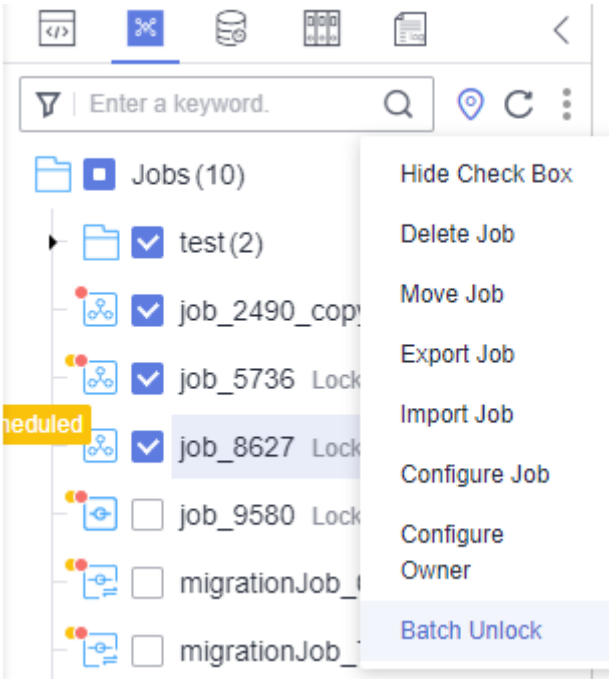
This section describes how to unlock jobs in batches.

Procedure

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. Click in the job directory and select **Show Check Box**.

5. Select the jobs to unlock, click , and select **Batch Unlock**. The message "Unlocked." is displayed.

Figure 6-70 Batch Unlock



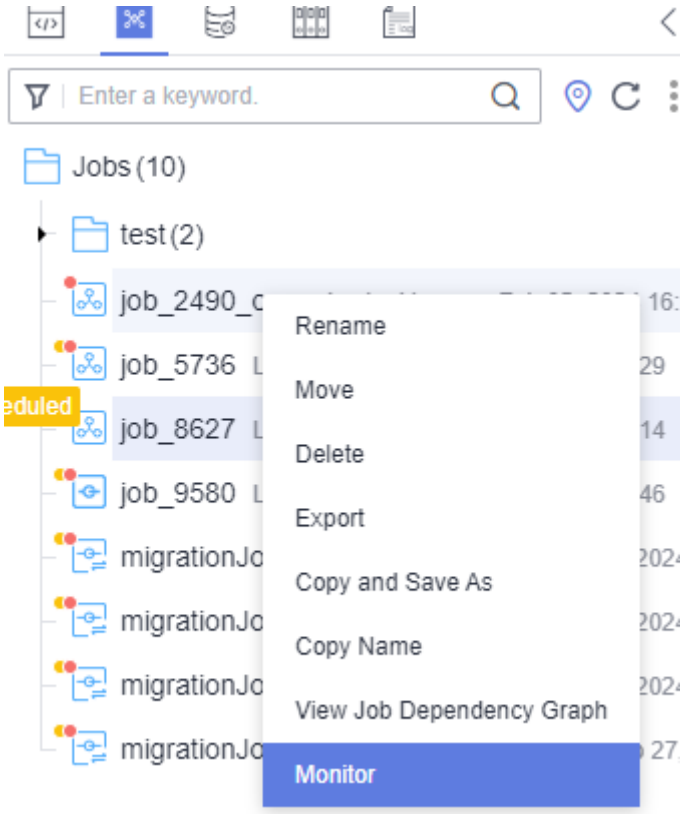
6.4.11.11 Going to Monitor Job page

From the job directory tree, you can quickly switch to the job monitoring page to view the monitoring details of the job.

Procedure

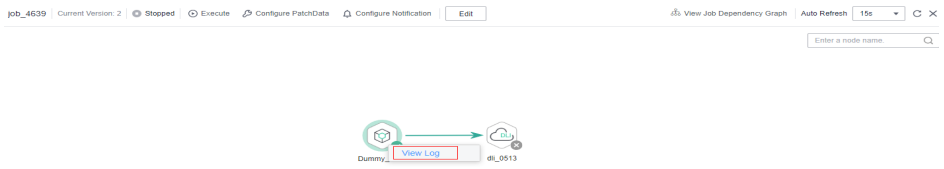
1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
4. Right-click a job in the directory tree and select **Monitor**.

Figure 6-71 Going to Monitor Job page



5. On the **Monitor Job** page, you can view the logs of the job nodes and the job version. You can also execute the job or click the job name or **Edit** to go to the job development page and modify job configurations.

Figure 6-72 Monitor Job page



6.5 Solution

Context

The solution aims to provide users with convenient and systematic management operations and better meet service requirements and objectives. Each solution can contain one or more business-related jobs, and one job can be used by multiple solutions.

You can perform the following operations on a solution:

- Creating a Solution
- Editing a Solution
- Exporting a Solution
- Importing a Solution
- Upgrading a Solution
- Deleting a Solution

Creating a Solution

On the development page of DLF, create a solution, set the solution name, and select business-related jobs.

1. Log in to the DataArts Studio console by following the instructions in Accessing the DataArts Studio Instance Console.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the navigation tree on the left of the data development page, choose **Development** > **Develop Script** or **Data Development** > **Develop Job**.
4. Above the directory on the left, click  to show the solution directory.
5. Click  in the upper part of the solution directory. The **Create Solution** page is displayed. Table 6-55 describes the solution parameters.

Figure 6-73 Creating a solution

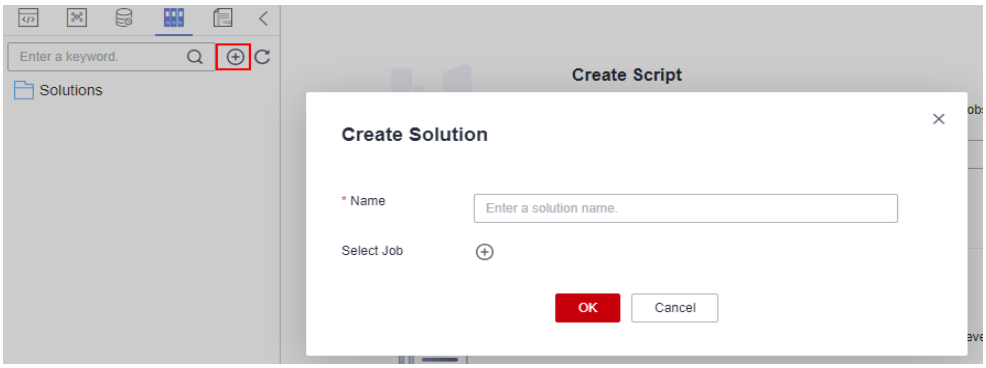


Table 6-55 Solution Parameters

Parameter	Description
Name	Name of the solution.
Select Job	Select the jobs contained in the solution.

6. Click **OK**. The new solution is displayed in the directory on the left.

Editing a Solution

In the solution directory, right-click the solution name and select **Edit** to change the name and job.

Exporting a Solution

In the solution directory, right-click the solution name and choose **Export** from the shortcut menu to export the solution file in ZIP format to the local host.

Importing a Solution

This solution is available only if the OBS service is available. If OBS is unavailable, data can be imported from the local PC.

In the solution directory, right-click a solution and choose **Import Solution** from the shortcut menu to import the solution file that has been uploaded to OBS or from a local directory.

NOTE

If you select **Overwrite** for **Duplicate Name Policy** but the hard lock policy is used and the script is locked by another user, the overwriting will fail. For details about soft and hard lock policies, see **Configuring the Hard and Soft Lock Policy**.

Upgrading a Solution

In the solution directory, right-click the solution name and choose **Upgrade** from the shortcut menu to import the solution file that has been uploaded to OBS. During the solution upgrade, the running jobs are stopped. The system determines whether to restart the jobs after the upgrade based on the configured upgrade restart policy.

Deleting a Solution

In the solution directory, right-click the solution name and choose **Delete** from the shortcut menu. A deleted solution cannot be restored. Exercise caution when performing this operation.

6.6 Execution History

This section describes how to view the execution history of scripts, jobs, and nodes over a week.

Prerequisites

This function depends on OBS buckets. For details about how to configure OBS buckets, see **Configuring an OBS Bucket**.

Script Execution History

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the navigation pane of the DataArts Factory homepage, choose **Data Development** > **Develop Script**.
3. Above the directory, click  to display the script and job execution history in the past seven days.

4. Select **Scripts** from the drop-down list box to filter out the script execution history.
5. Click a record to view the script information and execution result.
6. Download the historic script execution result.

   NOTE

   - By default, all users can download the historic execution results of scripts.
   - You can click **Download** on the **Result** tab page.
   - You can download the result file in CSV format. A maximum of 1,000 results can be queried and downloaded.

Job Execution History

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Job**.
3. Above the directory, click  to display the script and job execution history in the past seven days.
4. Select **Jobs** from the drop-down list box to filter out the job execution history.
5. Click a record to view the job and log information.

   NOTE

   If only some nodes of the job were tested, the execution history only displays information and logs for these nodes.

6.7 O&M and Scheduling

6.7.1 Overview

Choose **Monitoring** > **Overview**. On the **Overview** page, you can view the statistics of job instances in charts. Currently, you can view seven types of statistics:

- Status
  - You can filter out your or all job instances by selecting **My Instances** or **All**, and filter out job instances of the current day by selecting **Today**.
  - You can filter out your or all job instances by selecting **My Instances** or **All**, and filter out job instances of the current day by selecting **Yesterday**.
  - You can filter out your or all job instances by selecting **My Instances** or **All**, and filter out job instances of the current day by selecting **Two days ago**.
  - You can filter out your or all job instances by selecting **My Instances** or **All**, and filter out job instances of the current day by selecting **Last 7 days**.
  - You can click a status to go to the **Monitor Instance** page and view details about all jobs in the status.

NOTE

- The statistics include the monitoring data of the instances of real-time jobs. When you click a status, you will not be redirected to the **Monitor Instance** page of real-time jobs. Instead, you can only view the details of the instances of batch jobs.
- By default, the system displays all job instances of the current day.

- Completed Tasks

  NOTE

  Successfully executed instances of the current day are collected once an hour. A task is a job node.

  – You can view the number of all types of job nodes successfully executed on the previous day, on the current day, and over the last seven days on average.
  – You can view the number of a specific type of job nodes successfully executed on the previous day, on the current day, and over the last seven days on average.

- Tasks

  NOTE

  Number of tasks (operators in jobs) started in five minutes. Data of 30 days is available.

  – You can filter the operators started on each day within 30 days.
  – You can view the curve of the number of **all** operators that have been started.
  – You can view the curve of the number of different types of operators that have been started.

- Running DLI Jobs/Queue CU Usage

  You can filter the number of running DLI jobs and the CU usage of a specified queue.

  NOTE

  - You can view data of the last seven days by default, and view data of one month at most.
  - You can only view data of non-default queues. You can click the name of a queue to pin the queue to top.

- Number of Jobs and Number of Tasks Scheduled Daily

  NOTE

  This area displays the trend of the total number of jobs in a long period and the number of tasks scheduled each day. A task indicates an operator in a job.

  Number of jobs: total number of batch processing jobs and real-time jobs

  Number of tasks scheduled daily: number of tasks scheduled everyday, including both real-time and offline tasks. The number is calculated based on the nodes that are successfully scheduled.

  – By default, the system displays the number of jobs and the number of tasks scheduled each day in one month. You can filter data by time range.

- Task Type Distribution

  You can view the number of job nodes of different types.

  NOTE

  A task indicates an operator in a job.

  The system collects statistics on the number of nodes in all submitted jobs, including real-time jobs and batch processing jobs.

- Top 100 in Instance Running Time
  - You can filter out the top 100 instances of yours or all users with the longest running duration by time and owner.
  - You can click a job name to go to the **Monitor Instance** page and view the job running details.
  - By default, the system displays top 100 job instances in one month.
- Top 100 in Instance Failed to Run
  - You can filter out the top 100 instances of yours or all users with the most failures by time and owner.
  - You can click a job name to go to the **Monitor Instance** page, view the logs of the failed job instances, and analyze the causes.
  - By default, the system displays top 100 job instances in one month.
- End of scheduling in the next week

  You can view the jobs that are expected to be completed in the following week, including their names, owners, and end time.

  NOTE

  - Jobs that are expected to be completed in two days or less are displayed in red.
  - Jobs that are expected to be completed in three to five days are displayed in orange.
  - Jobs that are expected to be completed in six to seven days are displayed in black.

6.7.2 Monitoring a Job

6.7.2.1 Monitoring a Batch Job

In the batch processing mode, data is processed periodically in batches based on the job-level scheduling plan, which is used in scenarios with low real-time requirements. This type of job is a pipeline that consists of one or more nodes and is scheduled as a whole. It cannot run for an unlimited period of time, that is, it must end after running for a certain period of time.

You can choose **Monitor Job** and click the **Batch Job Monitoring** tab to view the scheduling status, scheduling period, and start time of a batch job, and perform the operations listed in **Table 6-56**.

Figure 6-74 Monitoring a Batch Job

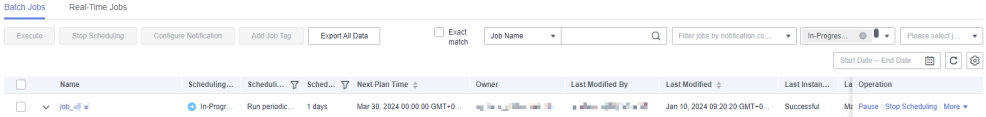


Table 6-56 Operations supported by batch job monitoring

No.	Operation	Description
1	Filtering jobs by **Job Name**, **Owner**, **CDM Job**, or **Node Type**	N/A
2	Filtering jobs by whether notifications have been configured, scheduling status, job tag, or next plan time	You can filter jobs for which no notification has been configured by notification type (such as exception or failure) so that you can set alarm notifications in batches.
3	Performing operations on jobs in a batch	Select multiple jobs and perform operations on them.
4	Viewing job instance status	Click in front of the job name. The **Last Instance** page is displayed. You can view information about the last instance of the job.
5	Viewing node information of the job	Click a job name. On the displayed page, click the job node and view its associated jobs/scripts and monitoring information. Click a job name. On the displayed page, view the job instance. For details, see **Batch Job Monitoring: Job Instances**.
6	Job scheduling operations	You can run, pause, recover, stop, and configure scheduling. For details, see **Batch Job Monitoring: Scheduling a Job**.
7	Configuring notifications	In the **Operation** column of a job, choose **More** > **Set Notification**. In the displayed dialog box, configure notification parameters. **Table 6-66** describes the notification parameters.
8	Monitoring instances	In the **Operation** column of a job, choose **More** > **Monitor Instance** to view the running records of all instances of the job.
9	PatchData	In the **Operation** column of a job, choose **More** > **PatchData**. For details, see **Batch Job Monitoring: PatchData**.
10	Adding a job tag	In the **Operation** column of a job, choose **More** > **Add Job Tag**. For details, see **Batch Job Monitoring: Adding a Job Tag**.

No.	Operation	Description
11	Viewing a job dependency graph	In the **Operation** column of a job, choose **More** > **View Job Dependency Graph**. For details, see **Batch Processing: Viewing a Job Dependency Graph**.
12	Exporting all data	Click **Export All Data**. In the displayed **Export All Data** dialog box, click **OK**. After the export is complete, go to the **Download Center** page to view the exported data. If the default storage path is not configured, you can set a storage path and select **Set as default OBS path** in the **Export to OBS** dialog box. A maximum of 30 MB data can be exported. If there are more than 30 MB data, the data will be automatically truncated. The exported job instances map job nodes. You cannot export data by selecting job names. Instead, you can select the data to be exported by setting filter criteria.

Click a job name. On the displayed page, view the job parameters, properties, and instances.

Click a node of a job to view the node properties, script content, and node monitoring information.

In addition, you can view the current job version and job scheduling status, schedule, stop, or pause a job, configure patch data, notification, or update frequency for a job.

Batch Job Monitoring: Job Instances

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Monitoring** > **Monitor Job**.
4. Click the **Batch Job Monitoring** tab.
5. Click a job name. On the displayed page, click the **Job Instances** tab to view job instances. You can perform the following operations:
   - Select **Show Instances to Be Generated** and set the time range to filter job instances that are expected to be generated in the future.

     NOTE

     A maximum of 100 instances expected to be generated can be displayed.

- Freeze or unfreeze job instances that are expected to be generated in the future. You can click **Freeze** or **Unfreeze** above the job instance list, or click **More** in the **Operation** column and select **Freeze** or **Unfreeze**.

  NOTE

  Freeze: You can only freeze job instances that have not been generated or are in waiting state.

  You cannot freeze jobs instances that have been frozen.

  When a job is frozen, it is considered to be failed and its downstream jobs will be suspended, executed, or canceled based on the failure policy configured for the job.

  When job instances that have not been generated are frozen, you can view them on the **Batch Job Monitoring** page or filter them by status on the **Monitor Instance** page.

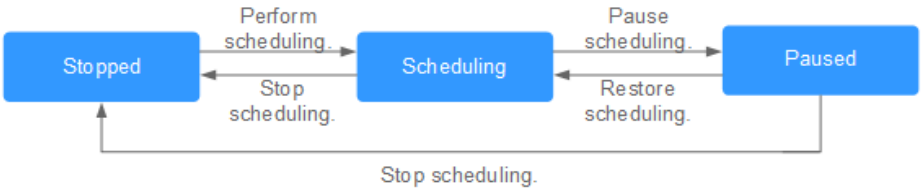
  Unfreeze: You can unfreeze a job instance that has not been scheduled and has been frozen.

- Perform other operations on job instances, such as stopping, rerunning, and retrying job instances, continuing running job instances, making job instances succeed, viewing waiting job instances, and viewing job configuration. When viewing waiting job instances, you can click **Remove Dependency** in the **Operation** column to remove dependency on an upstream instance.
- If jobs need manual confirmation before they are executed, they are in waiting confirmation state on the **Batch Jobs** page. When you click **Execute**, the jobs are in waiting execution state.

Batch Job Monitoring: Scheduling a Job

After developing a job, you can manage job scheduling tasks on the **Monitor Job** page. Specific operations include to run, pause, restore, or stop scheduling.

Figure 6-75 Scheduling a job



1. Log in to the DataArts Studio console by following the instructions in [Accessing the DataArts Studio Instance Console].
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Monitoring** > **Monitor Job**.
4. Click the **Batch Job Monitoring** tab.

   NOTE

   You can filter batch processing jobs by scheduling type or scheduling frequency.

5. In the **Operation** column of the job, click **Execute**, **Pause**, **Restore**, or **Stop Scheduling**.

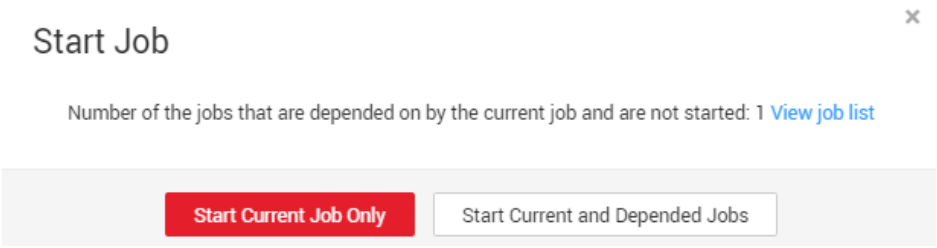
If a dependent job has been configured for a batch job, you can select either **Start Current Job Only** or **Start Current and Depended Jobs** when submitting the batch job. For details about how to configure dependent jobs, see **Setting Up Scheduling for a Job Using the Batch Processing Mode**.

NOTE

If the job is on the baseline task link, the system automatically displays a dialog box indicating that the baseline is associated when the scheduling is paused or stopped.

If the job is on the baseline task link or is depended on by other jobs, the system automatically displays a dialog box when the scheduling is paused or stopped.

Figure 6-76 Starting a job



Batch Job Monitoring: PatchData

A job executes a scheduling task to generate a series of instances in a certain period of time. This series of instances are called PatchData. PatchData can be used to fix the job instances that have data errors in the historical records or to build job records for debugging programs.

Only the periodically scheduled jobs support PatchData. For details about the execution records of PatchData, see **Monitoring PatchData**.

NOTE

Do not modify the job configuration when PatchData is being performed. Otherwise, job instances generated during PatchData will be affected.

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Monitoring** > **Monitor Job**.
4. Click the **Batch Job Monitoring** tab.
5. In the **Operation** column of the job, choose **More** > **Configure PatchData**.
6. Configure PatchData parameters based on **Table 6-57**.

Figure 6-77 PatchData parameters

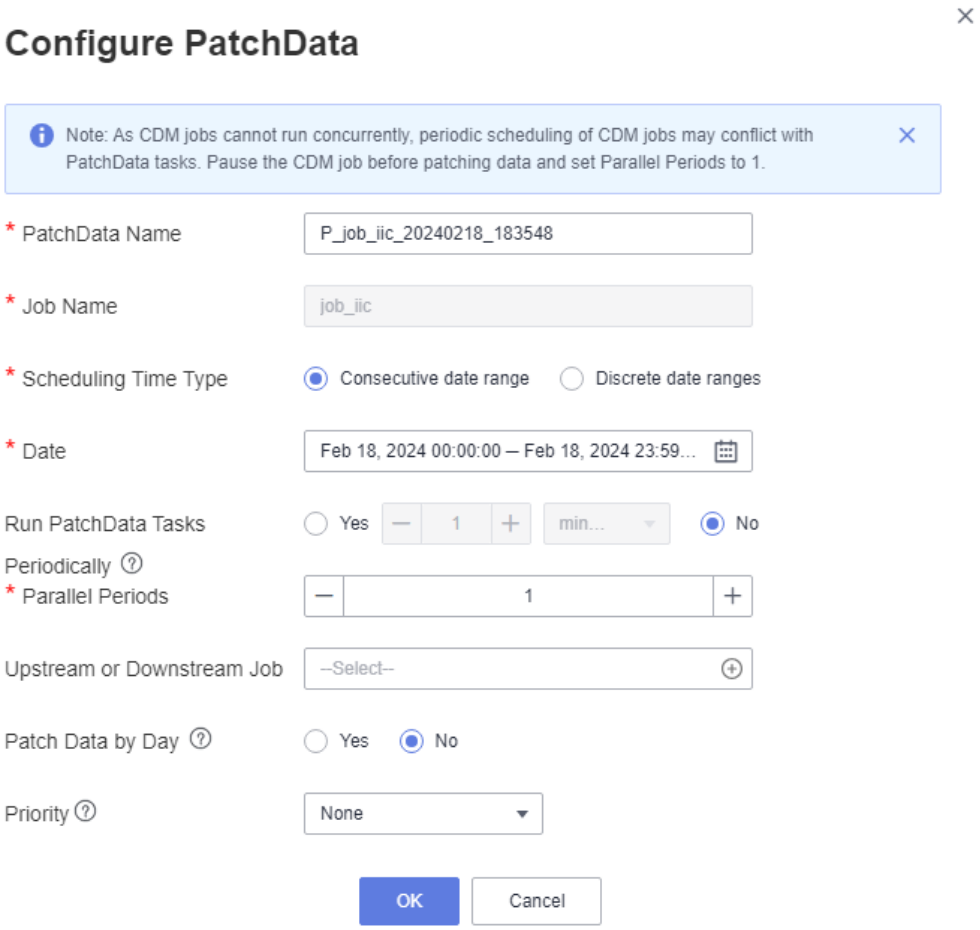


Table 6-57 Parameters

Parameter	Description
PatchData Name	Name of the automatically generated PatchData task. The value can be modified.
Job Name	Name of the job that requires PatchData.
Scheduling Time Type	• Consecutive date range The PatchData time is a continuous date range. • Discrete date ranges The PatchData time consists of discrete date ranges.

Parameter	Description
Date	**If Scheduling Time Type is set to Consecutive date range:** Period of time when PatchData is required. If the date is later than the current time, the current time is displayed by default. **NOTE** PatchData can be configured for a job multiple times. However, avoid configuring PatchData multiple times on the same date to prevent data duplication or disorder. If you select **Patch data in reverse order of date**, the patch data of each day is in positive sequence. **NOTE** • This function is applicable when the data of each day is not coupled with each other. • The PatchData job will ignore the dependencies between the job instances created before this date. **If Scheduling Time Type is set to Discrete date ranges:** You also need to set the following PatchData parameters: You can click **Add Date Range** to add multiple discrete date ranges for PatchData. You must set at least one date range. You can click **Delete** to delete discrete date ranges.

Parameter	Description
Run PatchData Tasks Periodically	• **Yes**: PatchData jobs will be executed based on the configured period. **The first value** indicates a specific value. **The second value** indicates that data is patched based on a specified period, for example, minutes, hours, days, weeks, or months. **NOTE** If you set a period, PatchData tasks will be scheduled based on that period. If the job is scheduled every few minutes, hours, or days, PatchData tasks will be scheduled based on the period you set. For example, if you want to patch data from 00:00 on Jan 1, 2023 to 00:00 on Feb 1, 2023 for an hourly job that starts at 01:00 every day, and set the PatchData period to two days, PatchData tasks will be scheduled at 00:00 on Jan 1, 2023, 00:00 on Jan 3, 2023, 00:00 on Jan 5, 2023, and so on. If the PatchData task scheduling period is in months and the first scheduling date falls on the last day of a month, PatchData tasks will be scheduled on the last day of each month. • **No**: PatchData jobs will not be executed periodically. Instead, the system executes PatchData jobs based on the existing rule.
Cycle	This parameter is required when **Scheduling Time Type** is set to **Discrete date ranges**. It specifies the PatchData cycle. You can click **Viewing Scheduling Details** to view the execution time of the task instances in the current time segment. **NOTE** This parameter is required only when a job is scheduled by hour or minute and **Scheduling Time Type** is set to **Discrete date ranges**.
Parallel Instances	Number of instances to be executed at the same time. A maximum of five instances can be executed at the same time. If you select **Yes** for **Patch Data by Day**, **Parallel Instances** means the number of concurrent job instances on the same day. If you select **No** for **Patch Data by Day**, **Parallel Instances** means the number of concurrent job instances in the scheduling cycle. **NOTE** Set this parameter based on the site requirements. For example, if a CDM job instance is used, data cannot be supplemented at the same time. The value of this parameter can only be set to **1**.

Parameter	Description
Upstream or Downstream Job	Select the upstream and downstream jobs (jobs that depend on the current job) that require PatchData. The job dependency graph is displayed. For details about the operations on the job dependency graph, see **Batch Processing: Viewing a Job Dependency Graph**.
Patch Data by Day	If you select **Yes**, PatchData instances on the same day can be executed concurrently for a job, but those on different days cannot be executed concurrently. For example, a job instance scheduled at 5:00 and one scheduled at 6:00 can be executed concurrently, but a job instance scheduled on 1st of a month and one scheduled on 2nd of the month cannot be executed concurrently. **Yes**: Data is patched by day. **No**: Data is not patched by day.
Stop Upon Failure	This parameter is mandatory if **Patch Data by Day** is set to **Yes**. **Yes**: If a daily PatchData task fails, subsequent PatchData tasks stop immediately. **No**: If a daily PatchData task fails, subsequent PatchData tasks continue. **NOTE** If data is patched by day and a PatchData task fails on a day, no PatchData task will be executed on the next day. This function is supported only by daily PatchData tasks, and not by hourly PatchData tasks.
Priority	Select a PatchData priority. You can set the priority of a workspace-level PatchData job in **Default Configuration**. **NOTE** The priority of PatchData is higher than that of PatchData in the workspace. Currently, only the priorities of DLI SQL operators can be set.
Ignore OBS Listening	• **Yes**: OBS listening is ignored in PatchData scenarios. • **No**: The system listens to the OBS path in PatchData scenarios.

Parameter	Description
Set Running Period	Whether a running period can be set for the PatchData task. • Yes You can set the time period for running the PatchData task every day. • No

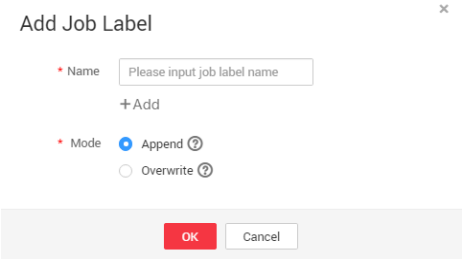
7. Click **OK**. The system starts to perform PatchData and the **PatchData Monitoring** page is displayed.

Batch Job Monitoring: Adding a Job Tag

Tags can be added to jobs to facilitate job instance filtering.

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Monitoring** > **Monitor Job**.
4. Click the **Batch Job Monitoring** tab.
5. In the **Operation** column of a job, choose **More** > **Add Job Tag**.
6. In the **Add Job Tag** dialog box displayed, set the job tag parameters.

   Figure 6-78 Parameters for adding a job tag

7. Click **OK**.

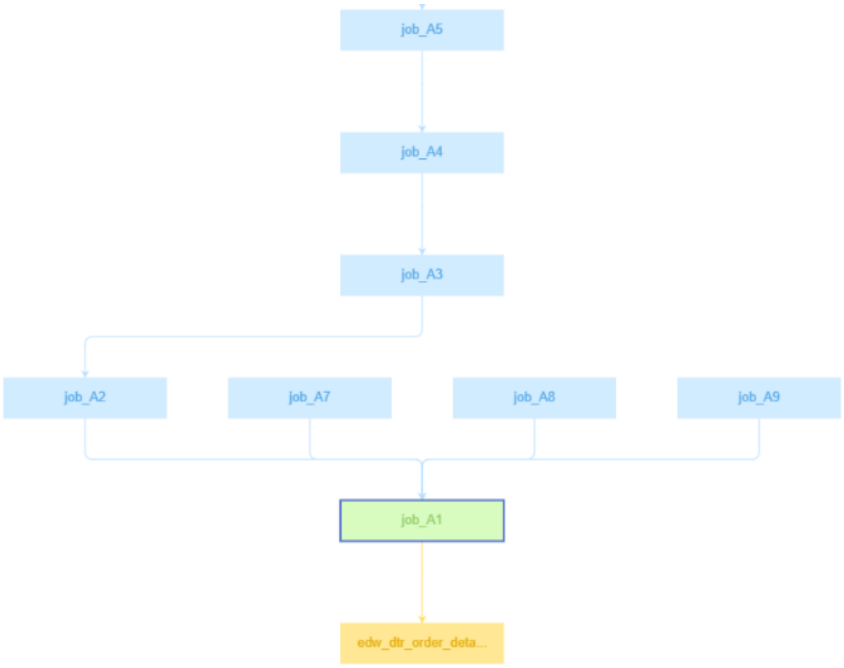
Batch Processing: Viewing a Job Dependency Graph

In the job dependency graph, you can view the dependencies between jobs.

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Monitoring** > **Monitor Job**.
4. Click the **Batch Job Monitoring** tab.

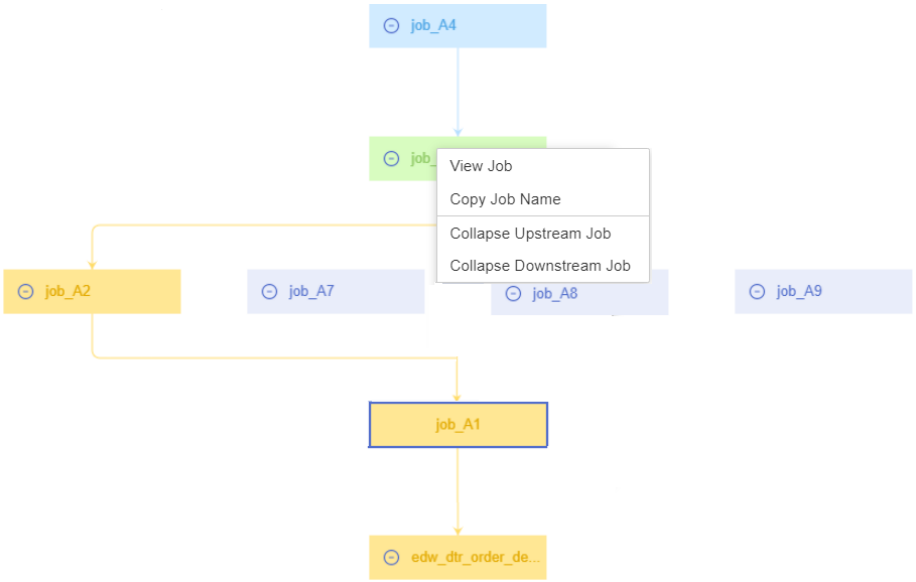
5. In the **Operation** column of a job, choose **More** > **View Job Dependency Graph**.
6. On the displayed **Job Dependency** page, perform any of the following operations:
   - In the upper right corner, select **Display complete dependency graphs**, **Display the current job and its upstream and downstream jobs**, or **Display the current job and its directly connected jobs**.
   - In the search box in the upper right corner, you can enter the name of a node to search for the node. The node found will be highlighted.
   - Click **Download** to download the job dependency file.
   - Scroll your mouse wheel to zoom in or zoom out the dependency graph.
   - Drag the blank area to view the complete relationship graph.
   - When the cursor is hovered on a job node, the node is marked green, its upstream job is marked blue, and its downstream job is marked orange.

     Figure 6-79 Marking upstream and downstream job nodes of a node

   - Right-click a job node to view the job, copy the job name, and collapse upstream or downstream jobs.

Figure 6-80 Job node operations



You can also view the node monitoring information of a job on the job details page.

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Monitoring** > **Monitor Job**.
4. Click the **Batch Job Monitoring** tab.
5. Click a job name and then a node to view monitoring information of the node.

   Click **Edit** to access the job development page.

6.7.2.2 Monitoring a Real-Time Job

In the real-time processing mode, data is processed in real time, which is used in scenarios with high real-time performance. This type of job is a pipeline that consists of one or more nodes. You can configure scheduling policies for each node, and the tasks started by nodes can keep running for an unlimited period of time. In this type of job, lines with arrows represent only service relationships, rather than task execution processes or data flows.

You can choose **Monitor Job** and click the **Real-Time Job Monitoring** tab to view the job status, start time, and end time, and perform the operations listed in **Table 6-58**.

Figure 6-81 Real-time job monitoring page

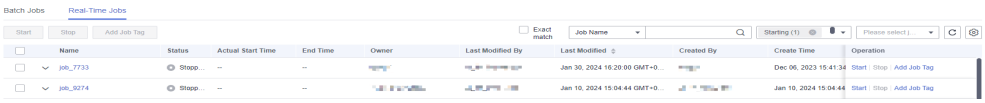


Table 6-58 Operations supported by real-time job monitoring

No.	Operation	Description
1	Filtering jobs by **Job Name**, **Owner**, **CDM Job**, or **Node Type**	N/A
2	Filtering jobs based on the job status or job tag	N/A
3	Perform operations on jobs in a batch	Select jobs and perform batch operations on them, including starting, stopping, and adding tags to them.
4	Viewing job instance status	Click job in front of the name. The **Last Instance** page is displayed. You can view information about the last instance of the job.
5	Job status-related operations	In the **Operation** column of a job, you can start, pause, recover, stop, rerun, and add tags to it.
6	Adding a job tag	Click **Add Job Tag**. The **Add Job Tag** dialog box is displayed.
7	Viewing node information of a job	Click a job name. On the displayed page, click a node to view its associated job/scripts and monitoring information. **NOTE** If event-driven scheduling is configured for a node in the job, the subjob monitoring page is displayed when you click the node.
8	Disabling and restoring a node	Click a job name. On the displayed page, right-click a node and select **Disable**. After the node is disabled, you can right-click it and select **Restore** to restore it on another location. For details, see **Real-Time Job Monitoring: Disabling and Restoring a Node**.
9	Viewing the boot log	Click a job name. On the displayed page, right-click a node and select **View Run Log** to view logs of the node.
10	Configuring scheduling	Click a job name. On the displayed page, right-click the node where event-driven scheduling is configured and select **Configure Scheduling** to modify the scheduling information about the node. For details, see **Real-Time Job Monitoring: Configuring Scheduling for a Node Where Event-driven Scheduling Is Configured**.

No.	Operation	Description
11	Clearing stream messages	Click a job name. On the displayed page, right-click the node where event-driven scheduling is configured and select **Clear Stream Message**.
12	Viewing logs	For real-time processing single-task Flink SQL and Flink JAR jobs, you can Click **More** and select **View Log** to view the logs of the jobs. **NOTE** This function is unavailable if the MRS cluster version is not supported.

Click a job name. On the displayed page, view the job parameters, properties, and instances.

Click a node of a job to view the node properties, script content, and node monitoring information.

In addition, you can view the current job version and status, start, rerun, and develop jobs, determine whether to display metric monitoring, and set the job refresh frequency.

Real-Time Job Monitoring: Disabling and Restoring a Node

You can disable a node in a real-time job and restore it in another location.

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Monitoring** > **Monitor Job**.
4. On the **Real-Time Job Monitoring** tab page, click a job name.
5. On the displayed page, right-click the node and select **Disable**.
6. Right-click the node and choose **Resume** from the shortcut menu. The **Resume Node Running** dialog box is displayed, as shown in **Table 6-59**.

Figure 6-82 Resuming node running

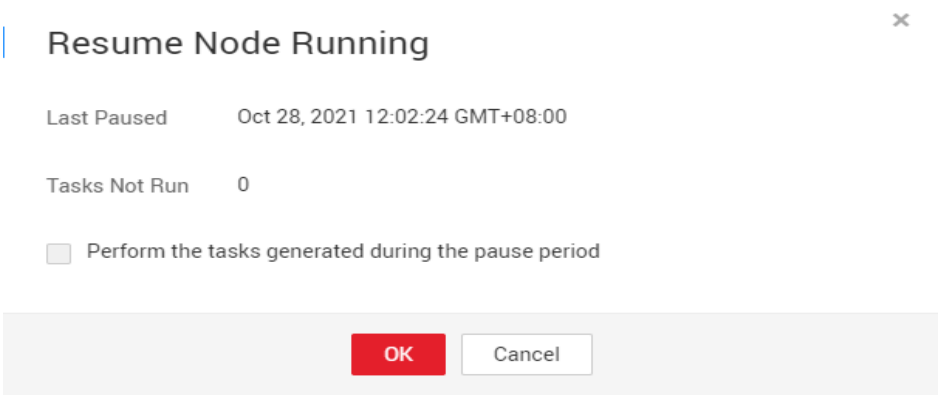


Table 6-59 Resumption parameters

Parameter	Description
Last Paused	Start time when a node is suspended.
Tasks Not Run	Number of tasks that are not running during node suspension.
Run From	Parameters for performing the tasks generated during the pause period. Position from which running restarts. • Paused node • The first node of the subjob
Concurrent Tasks	Parameters for performing the tasks generated during the pause period. Number of tasks to be processed.
Task Name	Parameters for performing the tasks generated during the pause period. Task to be resumed.

Real-Time Job Monitoring: Configuring Scheduling for a Node Where Event-driven Scheduling Is Configured

If event-driven scheduling is configured for a node in a real-time job, right-click the node on the job monitoring details page and choose **Configure Scheduling** from the shortcut menu to view and modify the scheduling information about the node.

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of DataArts Factory, choose **Monitoring** > **Monitor Job**.
4. On the **Real-Time Job Monitoring** tab page, click a job name.
5. On the displayed page, right-click the node where event-driven scheduling is configured, select **Configure Scheduling**, and configure the parameters shown in **Table 6-60**.

Figure 6-83 Configuring scheduling

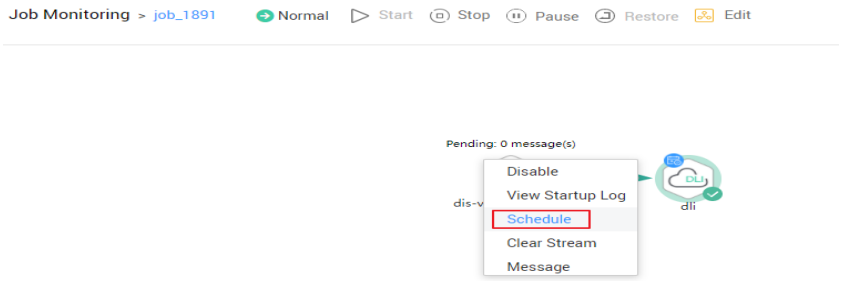
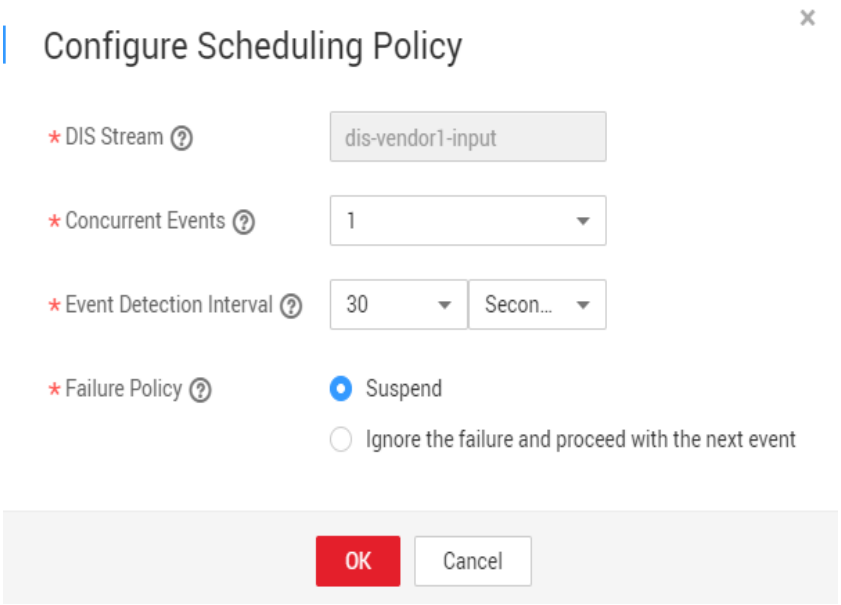


Table 6-60 Policy parameters

Parameter	Description
DIS Stream	Name of the DIS stream. When a new message is sent to the specified DIS stream, DataArts Factory transfers the new message to the job to trigger the job running.
Concurrent Events	Number of jobs that can be concurrently processed. The maximum number of concurrent events is 10.
Event Detection Interval	Interval for event detection. The unit of the interval can be **Seconds** or **Minutes**.
Failure Policy	Select a policy to be performed after scheduling fails. • Stop scheduling • Ignore failure and proceed

Figure 6-84 Configuring a DIS scheduling policy



6.7.3 Instance Monitoring

Each time a job is executed, a job instance record is generated. In the navigation pane of the DataArts Factory console, choose **Monitoring**. On the Monitor Instance page, you can view the job instance information and perform more operations on instances as required.

You can search for instances by **Job Name**, **Created By**, **Owner**, **CDM Job**, **Node Type**, and **Job Tag**. Search by CDM job is to search for job instances by node.

Performing Job Instance Operations

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the navigation pane on the left of the DataArts Factory page, choose **Monitoring** > **Monitor Instance**
4. You can stop, rerun, or continue to execute instances or make instances succeed. For details, see **Table 6-61**.

   When multiple instances are rerun in batches, the sequence is as follows:

   - If a job does not depend on the previous schedule cycle, multiple instances run concurrently.
   - If jobs are dependent on their own, multiple instances are executed in serial mode. The instance that first finishes running in the previous schedule cycle is the first one to rerun.
5. **Table 6-61** describes the operations that can be performed on the instance.

Table 6-61 Instance monitoring operations

Operation	Description
Searching for jobs by **Job Name**, **Created By**, or **Owner**	If you select **Exact search**, exact search by job name is supported. If you do not select **Exact search**, fuzzy search by job name is supported.
Filtering jobs by **CDM Job** or **Node Type**	N/A
Stop	Stop an instance that is in the **Waiting**, **Running**, or **Abnormal** state.

Operation	Description
Rerun	Rerun a subjob instance that is in the **Succeed** or **Canceled** state. For details, see **Rerunning Job Instances**. **NOTE** Manually scheduled jobs cannot be rerun. If instances need manual confirmation before they are executed, they are in waiting confirmation state when they are being rerun. When you click **Execute**, the instances are in waiting execution state.
Manual Retry	Retry abnormal instances.
Continue	Continue to run subsequent nodes in instances which are in abnormal state.
Succeed	Change the statuses of instances in **Abnormal**, **Canceled**, or **Failed** state to **Forcibly successful**.
Confirm Execution	Confirm executing instances in pending confirmation state.
More > Manual Retry	Retry abnormal instances.
More > View Waiting Job Instance	When the instance is in the waiting state, you can view the waiting job instance. Click **Remove Dependency** in the **Operation** column to remove dependency on an upstream instance.
More > Confirm Execution	Confirm executing instances in pending confirmation state.
More > Continue	If an instance is in the **Abnormal** state, you can click **Continue** to begin running the subsequent nodes in the instance. **NOTE** This operation can be performed only when **Failure Policy** is set to **Suspend the current job execution plan**. To view the current failure policy, click a node and then click **Advanced Settings** on the **Node Properties** page.
More > Succeed	Forcibly change the status of an instance from **Abnormal**, **Canceled**, or **Failed** to **Succeed**.
More > View	Go to the job development page and view job information.
More > History performance	You can view the historical performance of a job instance.
DAG	Display the DAG so that you can view the dependency between instances and perform O&M operations on the DAG. For details, see **Viewing the DAG**.

Operation	Description
Export All Data	Click **Export All Data**. In the displayed **Export All Data** dialog box, click **OK**. After the export is complete, go to the **Download Center** page to view the exported data. If the default storage path is not configured, you can set a storage path and select **Set as default OBS path** in the **Export to OBS** dialog box. A maximum of 30 MB data can be exported. If there are more than 30 MB data, the data will be automatically truncated. The exported job instances map job nodes. You cannot export data by selecting job names. Instead, you can select the data to be exported by setting filter criteria.

6. Click in front of an instance. The running records of all nodes in the instance are displayed.
7. Table 6-62 describes the operations that can be performed on the node.

Table 6-62 Operations (node)

Operation	Description
View Log	View the log information of a node.
Manual Retry	Retry a failed node. Retry an abnormal node. **NOTE** This operation can be performed only when **Failure Policy** is set to **Suspend the current job execution plan**. To view the current failure policy, click a node and then click **Advanced Settings** on the **Node Properties** page.
Succeed	Change the status of a node from **Failed** to **Succeed**. **NOTE** This operation can be performed only when **Failure Policy** is set to **Suspend the current job execution plan**. To view the current failure policy, click a node and then click **Advanced Settings** on the **Node Properties** page.
More > Skip	To skip a node that is to be run or that has been paused, click **Skip**. **NOTE** Instance with only one node cannot be skipped. Only instances with multiple nodes can be skipped.
More > Pause	When a job instance is in running state and a node is in waiting execution state, you can pause the node. Subsequent nodes will be blocked.

Operation	Description
More > Resume	To resume a paused node, click **Resume**.
More > History performance	You can view the historical performance of a job node.

Rerunning Job Instances

You can rerun a job instance that is successfully executed or fails to be executed by setting its rerun position.

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the navigation pane on the left of the DataArts Factory page, choose **Monitoring** > **Monitor Instance**
4. In the **Operation** column of a job, click **Rerun** to rerun the job instance. Alternatively, click the check box on the left of a job, and then click the **Rerun** button to rerun the job instance.

Figure 6-85 Setting the job rerunning

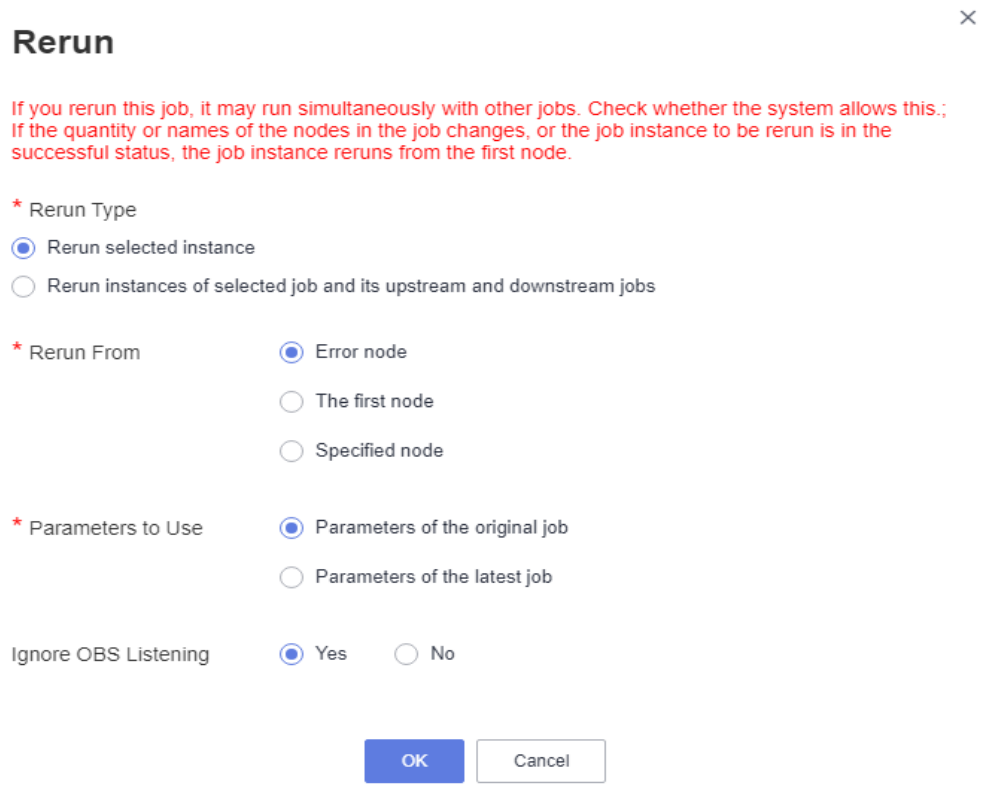


Table 6-63 Parameters for rerunning a job

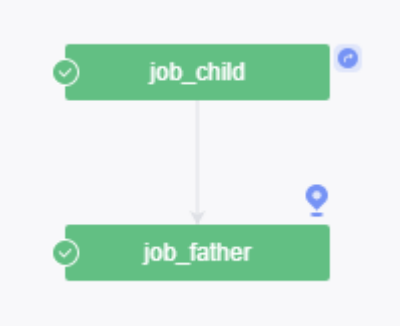
Parameter	Description
Rerun Type	Type of the instance that you want to rerun. • Rerun selected instance • Rerun instances of selected job and its upstream and downstream jobs
Start Time	Time range in which instances have been run
List of Rerun Job Instances	Upstream and downstream jobs to rerun. You can select multiple jobs at a time. The job dependency graph is displayed. For details about how to perform operations on the job dependency graph, see **Batch Processing: Viewing a Job Dependency Graph**.
Rerun From	Start position from which the job instance reruns. • **Error node**: When a job instance fails to be run, it reruns since the error node of the job instance. • **The first node**: When a job instance fails to be run, it reruns since the first node of the job instance. • **Specified node**: When a job instance fails to run, it reruns since the node specified in the job instance. This option is available only if **Rerun Type** is set to **Rerun selected instance**. **NOTE** A job instance reruns from its first node if either of the following cases occurs: • The quantity or name of a node in the job changes. • The job instance has been successfully run.
Parameters to Use	• Parameters of the original job • Parameters of the latest job
Concurrent Instances	Number of job instances that can be concurrently processed.
Ignore OBS Listening	• **Yes**: The system does not listen to the OBS path when rerunning the job instance. • **No**: The system listens to the OBS path when rerunning the job instance.

Viewing the DAG

You can view the dependency between instances and perform O&M operations on the DAG.

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the navigation pane on the left of the DataArts Factory page, choose **Monitoring** > **Monitor Instance**
4. Locate the row that contains a job and click **DAG** in the **Operation** column.

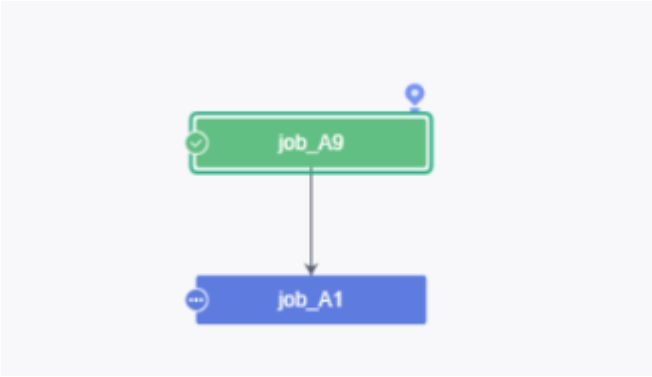
Figure 6-86 DAG



By default, the DAG displays the current job instance and its upstream and downstream job instances. It supports the following operations:

- Click  in the upper right corner of the DAG to restore the DAG to the initial state, and click  to close the DAG. Drag  in the upper left corner of the DAG to change its width.
- Click a job instance to select it.

Figure 6-87 Selecting a job instance

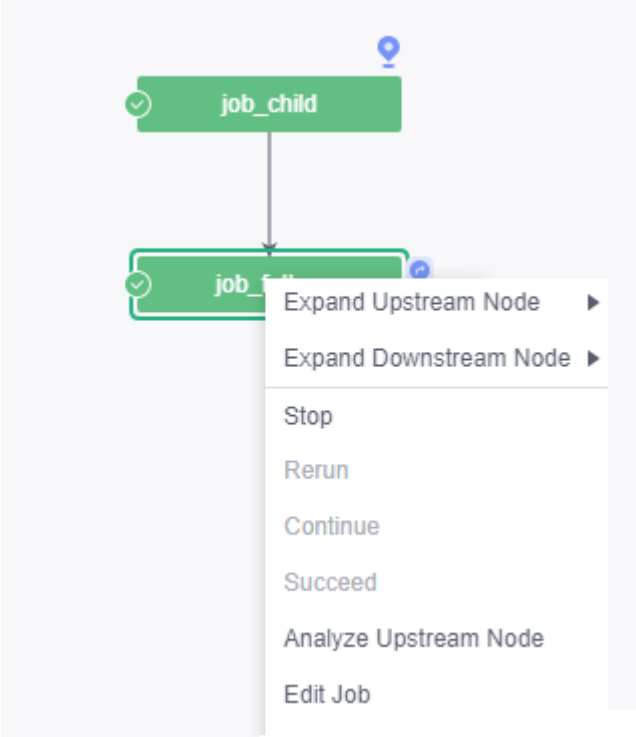


  - When a job instance is selected, the background colors of the job instance and its upstream and downstream instances are darkened.

- Brief information about the instance is displayed in the lower right corner of the DAG. The instance name and ID can be directly copied.
- Click **Show Details** to open the details panel, which displays information such as the instance attributes, job parameters, node list, and historical instances. You can adjust the height of the panel or close it.
- Click the blank area to deselect the job instance.

– Right-click a job instance to expand its upstream and downstream job instances. You can stop, rerun, continue to execute instances, forcibly make instances succeed, analyze the upstream node, and edit the job.

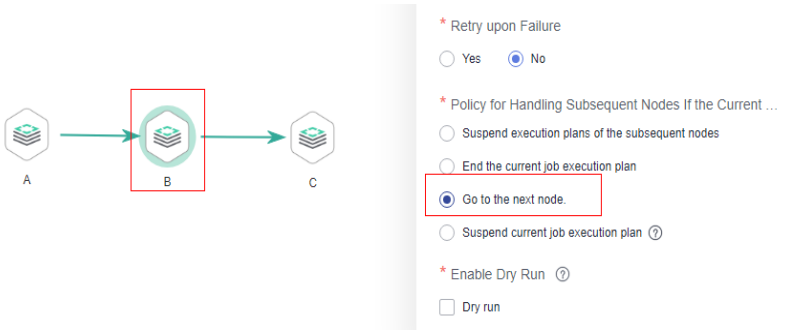
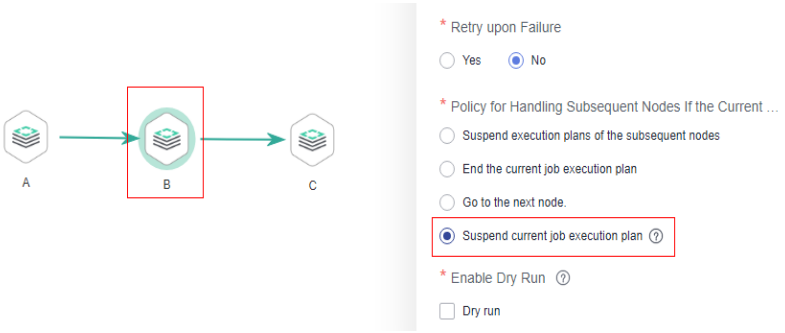
Figure 6-88 Performing operations on job instances



Job Instance Statuses

Table 6-64 Job instance statuses

Status	Description
Waiting	A job instance is in waiting state if the execution of its dependent job instances is not complete, for example, no instance has been generated, instances are waiting to be executed, or instances fail to be executed.
Running	A job is running. All of its dependent jobs have been executed successfully.

Status	Description
Successful	The service logic of a job is successfully executed (including the success of retry upon failure).
Forcibly successful	A job instance in failed or canceled state is made successful.
Failure ignored	As shown in the following figure, a failure handling policy is configured to skip node B and continue to execute node C if node B fails. When the job is executed successfully, the job instance is in **Failure ignored** state. **Figure 6-89** Failure handling policy – Go to the next node 
Abnormal	There are few scenarios where this status is displayed. As shown in the following figure, a failure handling policy is configured to suspend the job instance immediately without continuing to execute node C. In this case, the job instance is in **Abnormal** state. **Figure 6-90** Failure handling policy – Suspend current job execution plan 
Paused	There are few scenarios where this status is displayed. When a running job instance is suspended by the test personnel, the instance is in **Paused** state.

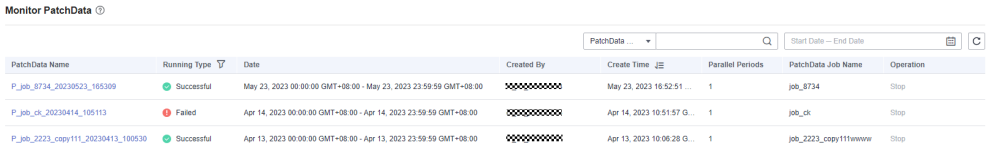
Status	Description
Canceled	• If you manually stop a job instance in **Waiting** state, the job instance status becomes **Canceled**. • If you stop scheduling the upstream job on which a job instance depends, the job instance status becomes **Canceled**. For example, job A depends on job B. If you stop scheduling job B, the instance generated for job A is automatically canceled.
Frozen	If a job instance is expected to be generated in the future, the job instance is in frozen state after being frozen.
Failed	A job fails to be executed.

6.7.4 Monitoring PatchData

In the navigation tree of the DataArts Factory console, choose **Monitoring** > **Monitor PatchData**.

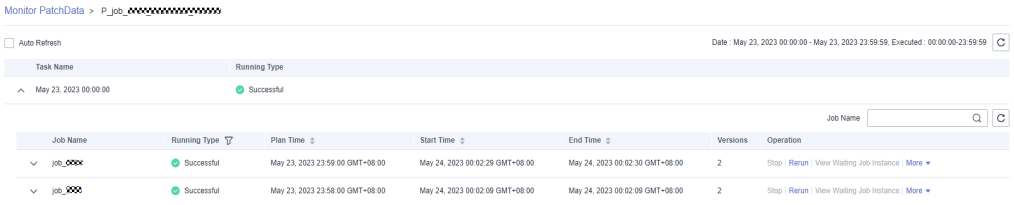
On the page shown in **Figure 6-91**, you can view the PatchData job status, date, number of parallel periods, PatchData job name, creator, creation time, and stop a running job. You can filter jobs by PatchData name, creator, date, and status.

Figure 6-91 PatchData Monitoring page



On the page shown in **Figure 6-91**, click PatchData name. On the displayed page, you can view the PatchData execution status. For more information, see **Batch Job Monitoring: PatchData**.

Figure 6-92 PatchData monitoring details



NOTE

- PatchData can be sorted by plan time, start time, and end time. Note that only one of the three sorting modes takes effect at a time.
- Click the sorting icon once to sort PatchData in ascending order, click the sorting icon twice to sort PatchData in descending order, and click the sorting icon three times to cancel sorting.
- When viewing a waiting job instance, click **Remove Dependency** in the **Operation** column to remove dependency on an upstream instance.
- If a PatchData task fails, you can click **Operation** and select **Stop** to stop the task.
- On the PatchData details page, you can perform a fuzzy search of PatchData jobs by job name.
- If job instances need manual confirmation before they are executed, they are in waiting confirmation state on the **Monitor PatchData** page. When you click **Execute**, the job instances are in waiting execution state.

6.7.5 Baseline O&M

6.7.5.1 Overview

Baseline O&M allows you to configure baseline tasks to monitor task statuses and resource usage. By configuring O&M baselines, you can ensure that important data is generated within the expected time in complex dependency scenarios. Baseline O&M effectively reduces configuration costs, avoids invalid alarms, and automatically monitors all important tasks.

Application scenarios:

- Managing task priorities

  When the number of tasks keeps increasing but resources are limited, you can add important tasks to the baseline and set a higher priority for the baseline so that resources are preferentially allocated to important tasks.

- Estimating the task completion time

  The running of tasks is affected by resources and their upstream tasks. You can add a task to the baseline, and the system will calculate the estimated completion time of the task.

- Assuring on-time task completion

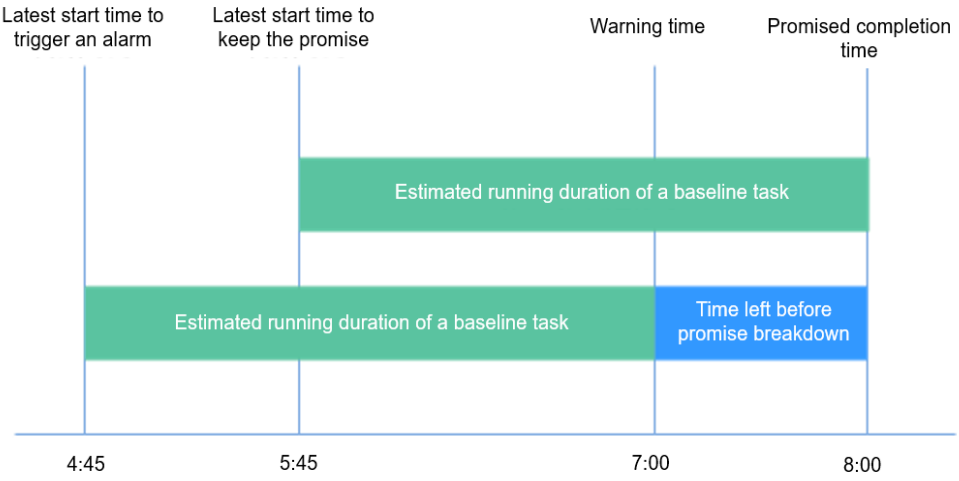
  You can add a task to a baseline and set a promised completion time. If the system predicts that the task cannot be completed before the promised time, or an upstream task is faulty or slows down, an alarm will be sent. You can handle the issue in a timely manner based on the alarm information to ensure that the task can be completed before the promised time.

Concepts

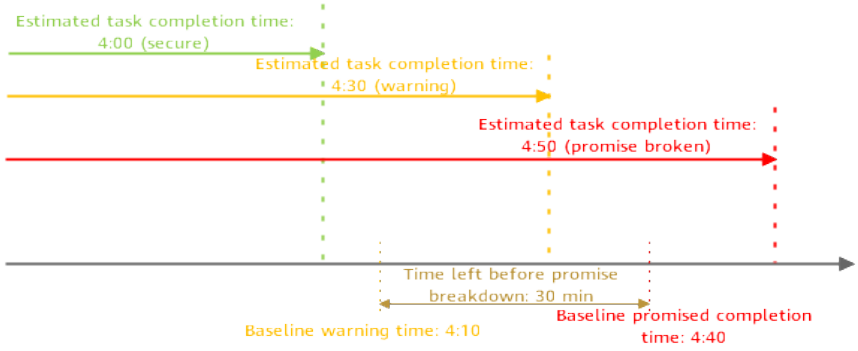
- Baseline: After you add an important task to the baseline and set a promised completion time, the system calculates the estimated completion time of the task based on the task running status. If the system determines that the task may not be completed before the promised time, the system generates an alarm.
- Promised completion time: indicates the latest time when a task should be successfully executed for data applications. If you want to reserve some time

for O&M personnel to handle exceptions, you can set a time left before promise breakdown. The system uses the promised completion time minus the time left before promise breakdown as the warning time for triggering an alarm.

- Time left before promise breakdown: A baseline warning is triggered at the time calculated by the promised completion time minus this time.
- Warning time: It equals the promised completion time minus the time left before promise breakdown.
- Estimated running duration: estimated running duration of the current task calculated based on the running duration of historical tasks
- Latest start time to keep the promise: promised completion time minus estimated task running duration
- Latest start time to trigger an alarm: warning time minus estimated task running duration



- Baseline task: a task added to a baseline
- Baseline instance: The system uses this instance to calculate the estimated completion time of each task. The status of the baseline instance can be secure, warning, or promise broken.



  - Secure: estimated completion time < warning time
  - Warning: warning time < estimated completion time < promised completion time

  - Promise broken: estimated completion time > promised completion time
- Key path: the path that takes the longest time to run among multiple paths that affect the baseline task
- Event: An event is generated when an error occurs in the baseline task or its upstream tasks, or when a task on the critical path becomes slow. Events affect the on-time completion of baseline tasks.

Monitoring Scope

Key tasks and all the upstream tasks on which the key tasks depend

Functions

After important tasks are added to a baseline, the system assures resources for the tasks based on the baseline priority, determines the monitoring scope based on the upstream and downstream dependencies of the baseline tasks, and triggers baseline alarms or event-based alarms based on the statuses of the monitored tasks. Baseline O&M provides the following functions:

- Alarms for failures of key tasks
- Alarms for delay of key tasks
- Key path analysis
- Preferential scheduling of key tasks
- Alarms for key tasks
- Immediate alarms for configuration errors
- Full-link version comparison for key jobs

Alarm Mechanism

A baseline alarm is an alarm notification for a baseline that is enabled and whose alarm function is enabled. You can configure the time left before promise breakdown and the promised completion time based on the estimated completion time of the baseline. The system calculates the estimated latest completion time of a monitored task based on the historical running status of the task and monitors the task based on the actual running status of the baseline task. If the system predicts that the baseline task cannot be completed before the baseline warning time (baseline promised completion time – time left before promise breakdown), the system sends a baseline alarm to the alarm recipients defined for the baseline.

Alarm Types

- Baseline warning

  First task in the monitored baseline link that is not completed before the warning time
- Baseline promise breakdown

  The baseline promise breaks down when the following conditions are met:

  a. No promise breakdown occurs in the direct or indirect upstream of the task node.

b. The task is not completed by the promised completion time.

- Further promise breakdown

  This alarm is triggered when the following conditions are met:

  a. An alarm has been triggered for the task.

  b. The task running time is longer than estimated. To be specific:

- Assured task not completed by warning time

  This alarm is generated when some assured tasks are not completed by the baseline warning time (promised completion time – time left before promise breakdown). Only one alarm is generated for an assured task.

- Assured task not completed by promised completion time

  This alarm is generated when some assured tasks are not completed by the baseline promised completion time. Only one alarm is generated for an assured task.

- Task failure

  This alarm is generated when any monitored task fails or stops being scheduled due to incorrect configurations.

6.7.5.2 Restrictions

When using the baseline O&M feature, pay attention to the following requirements to ensure that the task priority takes effect:

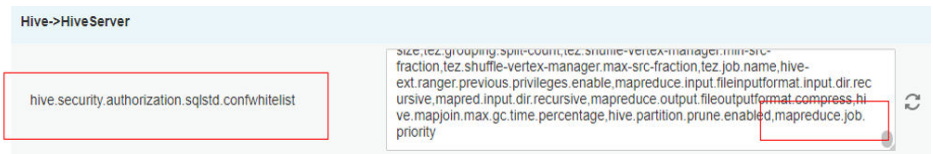
MRS restrictions:

- For MRS clusters, priority configuration is available for data connections in MRS API mode and unavailable for data connections in proxy mode.
- To enable priority for MRS nodes of DataArts Studio (if the MRS cluster is in security mode, that is, Kerberos authentication is enabled), add the whitelist configured for MRS Hive, configure the following parameters, and click **Save**. The configuration takes effect after a rolling restart of the parameter. Set this parameter in the basic configuration on the cluster O&M management page of MRS Hive. The operations are as follows:

  a. Log in to FusionInsight Manager and choose **Cluster** > **Services** > **Hive**. Click **Configurations** then **All Configurations**.

  b. In the navigation pane on the left, choose **Hive** > **HiveServer**. Add **mapreduce.job.priority** to the value of **hive.security.authorization.sqlstd.confwhitelist**.

  Figure 6-93 Configuring hive.security.authorization.sqlstd.confwhitelist

  c. Save the change and restart Hive.

NOTE

Priority configuration is available for the following MRS nodes: MRS Spark SQL, MRS Hive SQL, MRS Spark, MRS Flink Job, and MRS MapReduce.

To make the Hive priority take effect, contact O&M engineers to enable the MRS Hive priority configuration item.

- Before using baseline O&M, you must create a topic in MRS. For details, see **Creating a Topic on Kafka UI**.

DLI restrictions:

DLI allows you to set job priorities. When resources are insufficient, computing resources are preferentially provided for jobs with higher priorities. DLI priority configuration is available for DLI Flink Job, DLI SQL, and DLI Spark job operators.

NOTE

- Priorities can only be set for jobs running in elastic resource pools.
- SQL jobs in elastic resource pools support priorities.
- Priorities can be set for jobs of Spark 2.4.5 or later.
- Priorities can be set for jobs of Flink 1.12 or later.

6.7.5.3 Baseline Instances

The system uses this type of instances to calculate the estimated completion time of each task. This section describes how to view baseline instance details and baseline notifications.

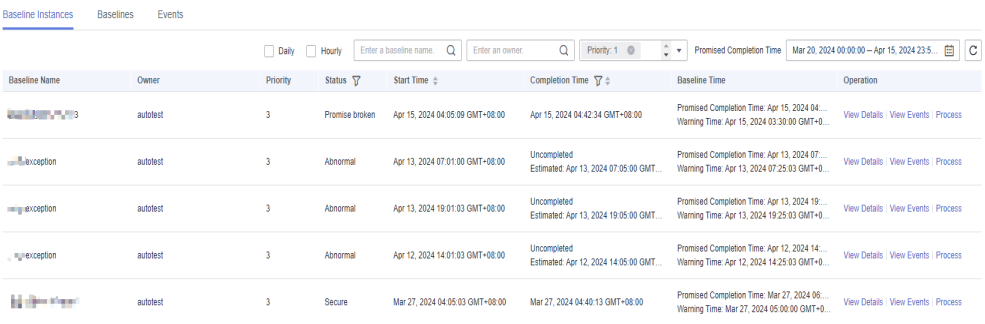
Restrictions

Baseline instances are generated only if the baseline is enabled. For details, see **Baseline Management**.

Viewing the Baseline Instance List

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the navigation pane on the DataArts Factory console, choose **Monitoring** > **Baseline O&M**.
3. Click the **Baseline Instances** tab.
4. In the baseline instance list, you can view details about baseline instances, including their names, owners, priorities, statuses, start time, completion time, and baseline time.

Figure 6-94 Baseline instance list



You can quickly find a desired baseline instance by name, owner, priority, or promised completion time. You can filter baseline instances by priority or promised completion time.

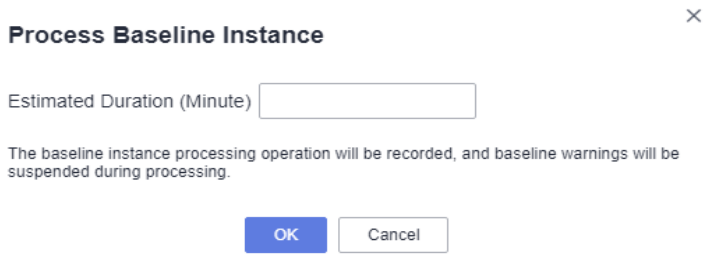
NOTE

Baseline statuses include:

- **Secure**: The task was completed by warning time.
- **Warning**: The task was not completed by warning time, but the promised completion time has not come.
- **Promise broken**: The task was not completed by promised completion time.
- **Abnormal**: All baseline tasks are suspended, or the baseline is not associated with any task.

5. You can view more details about a baseline instance, for example, events related to the instance.
   - Click **View Details** in the **Operation** column. On the **Baseline Instance Details** page, you can view the following information of the baseline instance: basic information, assured jobs, key path jobs and the Gantt chart, comparison of the versions before and after the baseline job modification, and related events.
   - Click **View Events** in the **Operation** column. On the displayed page, you can view the ID, description, owner, status, and the time when the event occurred.
   - Click **Process** in the **Operation** column. In the displayed **Process Baseline Instance** dialog box, set **Estimated Duration (Minute)** and click **OK**.

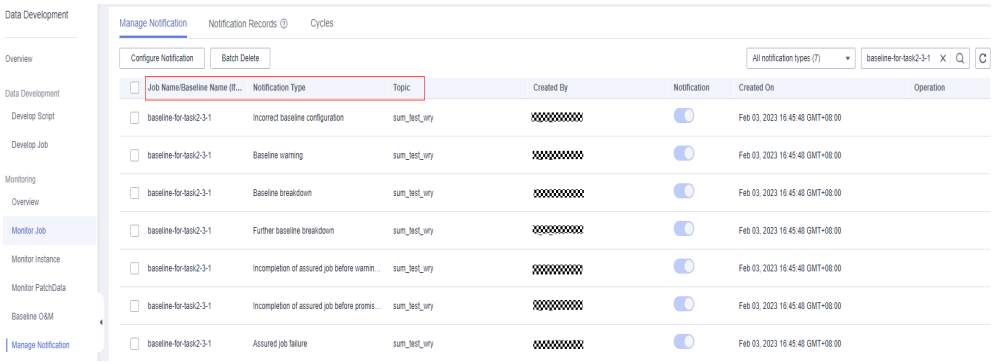
     The baseline instance processing operation will be recorded, and baseline alarms will be suspended during processing. The baseline owner does not send alarms during processing.



Viewing Baseline Notifications

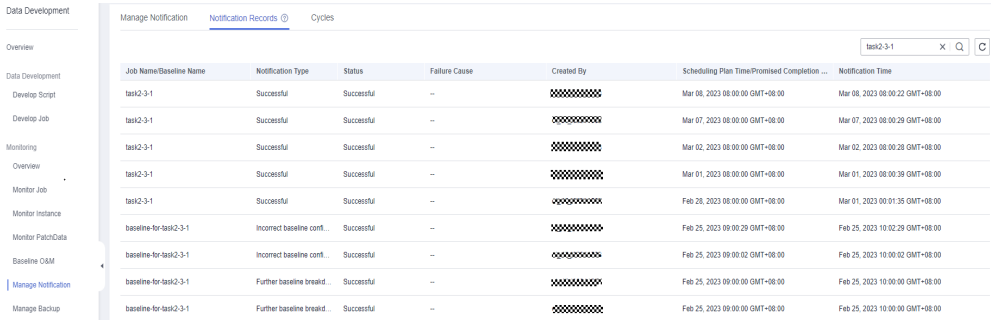
1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the navigation pane on the DataArts Factory console, choose **Monitoring** > **Manage Notification**.
3. On the **Manage Notification** tab page, you can view notifications of the baseline. Each notification contains the baseline name, notification type, and topic.

Figure 6-95 Notification list



4. Click the **Notification Records** tab to view all notifications of the baseline

Figure 6-96 Baseline notifications



For details about notification management, see **Managing Notifications**.

6.7.5.4 Baseline Management

To ensure that important tasks can be completed on time, you can add the tasks to a baseline and set the promised completion time and time left before promise breakdown. When the system determines that the baseline task may not be completed before the promised time, the system generates an alarm.

Creating a Baseline

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the navigation pane on the DataArts Factory console, choose **Monitoring** > **Baseline O&M**.
3. Click the **Baselines** tab.
4. Click **Create**.
5. Set the baseline parameters shown in **Table 6-65**.

Figure 6-97 Creating a baseline

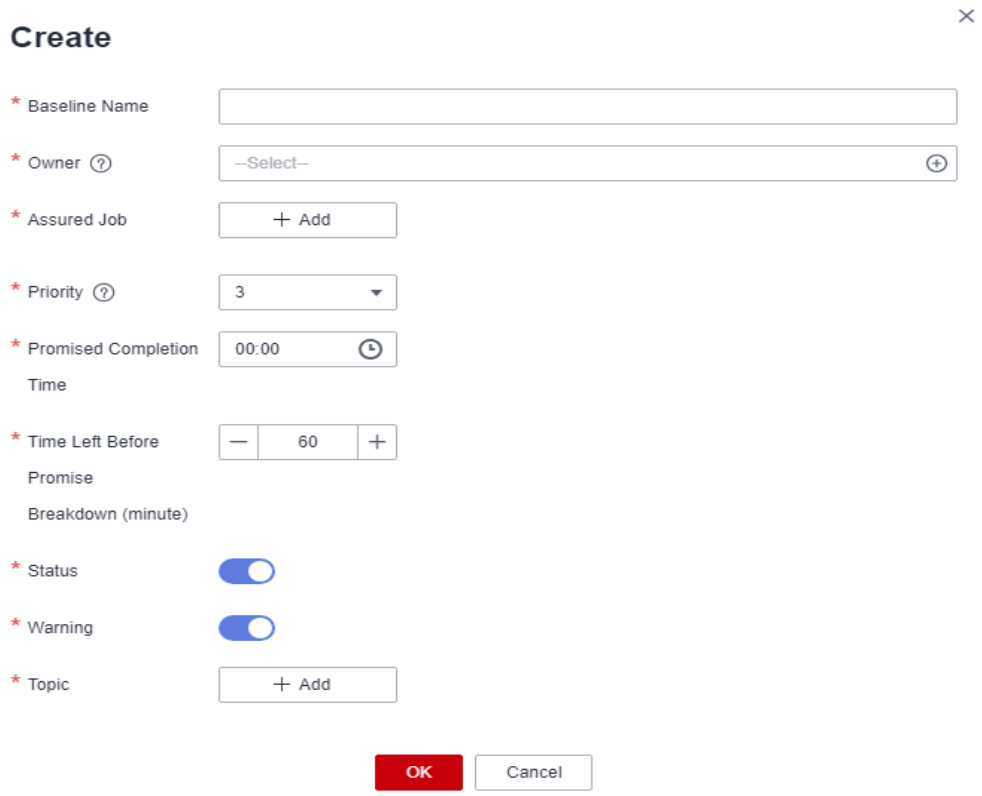


Table 6-65 Baseline parameters

Parameter	Description
Baseline Name	Enter the baseline name.
Owner	Select a baseline owner. **NOTE** To add or delete owners, go to the **Workspaces** page.
Assured Job	Select the jobs to be added to the baseline. Click **Add** and select assured jobs. You can search for keywords to quickly find your desired baseline assured jobs. You can delete selected jobs and add them again. **NOTE** You are advised to select downstream nodes of the business process. After they are selected, all the upstream nodes that affect the data output of the selected nodes will be monitored. You are not advised to add all tasks in the business process to the baseline.

Parameter	Description
Priority	Set the priority of the baseline. The following priorities are supported: • 1 • 2 • 3 • 4 • 5 **NOTE** A larger value indicates a higher priority for a baseline as well as the tasks added to the baseline. If resources are insufficient, tasks with higher priorities will be allocated scheduling resources first. The configured priority takes effect for the cycle instance generated on the next day. For MRS clusters, priority configuration is available for data connections in MRS API mode and unavailable for data connections in proxy mode. For more information about priority constraints, see **Restrictions**.
Promised Completion Time	• Set the promised completion time for the baseline task. The baseline calculates the warning time based on the promised completion time. Ensure that the promised completion time minus the time left before promise breakdown is later than the estimated completion time of the baseline task. **NOTE** Baseline warning time equals the promised completion time minus the time left before promise breakdown. If the actual completion time is later than the promised completion time minus the time left before promise breakdown, an alarm is triggered. For example, if the promised completion time is set to 04:30 and the time left before promise breakdown is set to 20 minutes, an alarm will be generated if the system estimates that the task cannot be completed by 04:10.
Time Left Before Promise Breakdown (minute)	Set the time left before the promised completion time. This parameter specifies the warning time for the baseline. The interval between the promised and estimated completion time must be at least 5 minutes. Otherwise, alarms will be frequently reported. You are advised to configure this parameter based on the running duration of baseline tasks.

Parameter	Description
Status	 : The system will monitor baseline tasks and all their dependent upstream tasks.  : The system will not monitor baseline tasks or any of their dependent upstream tasks.
Warning	 : If the system estimates that a baseline task cannot be completed by the promised completion time, or an upstream task is faulty or slows down, the system sends an alarm to you.  : You will not receive any alarms of baseline tasks.
Topic	Add SMN topics. Click **Add**. In the displayed dialog box, search for desired topics by keyword and select them. You can delete selected topics and add them again. If no topic is available, click **View Topic** to create one. For details, see **Creating a Topic**.

6. Click **OK**.
7. In the baseline list, you can view, query, modify, or delete baselines. The operations are as follows:
   - Query baselines by name, owner, or priority. You can filter baselines by priority, status, daily baseline, and hourly baseline.
   - Click **Edit** in the **Operation** column. In the displayed **Edit Baseline** dialog box, modify parameter settings of the baseline.
   - Click **Delete** in the **Operation** column. In the displayed **Delete This Baseline?** dialog box, click **OK** to delete the baseline.

6.7.5.5 Event Management

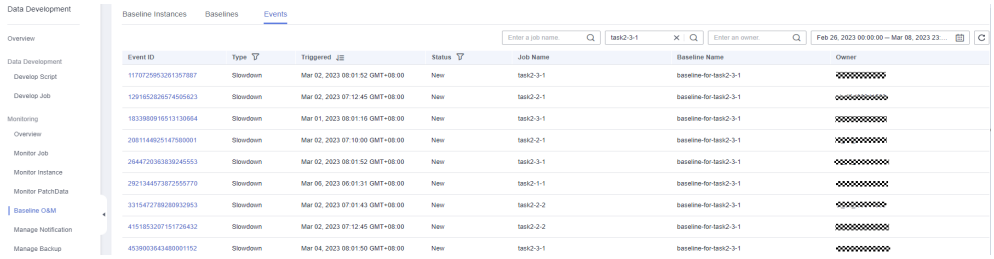
If an error occurs in a baseline task or its upstream task, or a task in the key path slows down, an event is generated. You can view the event details in **Baseline O&M** > **Events**. Baseline O&M can detect exceptions that cause task completion failures and generate alarms in advance to ensure that important data can be generated within the expected time in complex dependency scenarios.

Viewing the Event List

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the navigation pane on the DataArts Factory console, choose **Monitoring** > **Baseline O&M**.
3. Click the **Events** tab.

4. On the **Events** tab page, you can view the event details, including the event ID, type, status, job name, baseline name, owner, and the time when the event was triggered.

   Figure 6-98 Event list

   You can quickly find your desired events by job name, baseline name, owner, or time when events were triggered.

   NOTE

   The following event types are available:

   - **Error**: The task failed to be executed.
   - **Slowdown**: The current running duration of the task is much longer than its average running duration in a certain period of time in the past.

     If the execution of a task becomes slow and then encounters an error, two events will be generated.

   The following event statuses are available:

   - **New**: The monitored baseline task slowed down or failed.
   - **Restored**: The task was executed successfully, though later than the promised completion time.
   - **Processing**: The event generated by the baseline is being processed.
   - **Ignored**: The event generated by the baseline has been ignored.

5. Click **Event ID** to go to the **Event Details** page, where you can view event details.

6.7.5.6 Properly Configuring the Promised Completion Time and Time Left Before Promise Breakdown

This section describes how to configure the promised completion time and time left before promise breakdown.

- Baseline O&M can detect exceptions that cause task completion failures and generate alarms in advance to ensure that important data can be generated within the expected time in complex dependency scenarios.
- The promised completion time for a baseline indicates the latest time when a task should be successfully executed for data applications. If you want to reserve some time for O&M personnel to handle exceptions, you can set a time left before promise breakdown. The system uses the promised completion time minus the time left before promise breakdown as the warning time for triggering an alarm.
- For details about how to set the promised completion time and time left before promise breakdown, see **Baseline Management**.

Properly Configuring the Promised Completion Time and Time Left Before Promise Breakdown

You should set the promised completion time for a baseline based on the latest completion time of baseline tasks in a historical period to ensure that the promised completion time is later than this latest completion time and reserve the time left before promise breakdown. This ensures that you can handle exceptions by the promised completion time after receiving an alarm.

Example Scenarios Where the Promised Completion Time and Time Left Before Promise Breakdown Are Not Properly Configured

If the two values are not properly configured, the baseline promise may be broken. As a result, the baseline warning does not meet the expectation.

- Scenario 1: The promised completion time for the baseline is the same as the latest task completion time, and no time left before promise breakdown is configured for the baseline.

  When a task encounters an exception, promise breakdown may occur as no time left before promise breakdown is configured. As a result, baseline alarms are frequently reported.
- Scenario 2: The time left before promise breakdown is not properly configured, that is, the baseline warning time (promised completion time for the baseline – time left before promise breakdown) falls within the latest task completion period.

  When a task encounters an exception, promise breakdown may occur as the time left before promise breakdown is insufficient. As a result, baseline alarms are frequently reported.
- Scenario 3: The promised completion time for the baseline is not properly configured, that is, the promised completion time is before the latest task completion time for the baseline.

  Tasks cannot be completed by the promised completion time. Baseline alarms will be generated at 00:00 on the current day.

6.7.6 Managing Notifications

DataArts Studio uses Simple Message Notification (SMN) to send push notifications based on your subscription requirements, so that you can receive immediate notifications when a job encounters an exception or runs successfully.

6.7.6.1 Managing Notifications

You can configure job notification tasks to notify you of job success or failures.

Configuring a Notification

Before configuring a notification for a job:

- Message notification has been enabled and a topic has been configured.
- A job not in **Not Activated** status has been submitted.

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

2. In the navigation pane on the DataArts Factory page, choose **Monitoring** > **Manage Notification**.
3. On the **Notification Management** tab page, click **Configure Notification**. In the displayed dialog box, configure parameters. **Table 6-66** describes the parameters.

Table 6-66 Notification parameters

Parameter	Mandatory	Description
Notification Scope	Yes	Notification scope. Available options include: • **One job**: Notifications are sent for a single job. • **All jobs**: Notifications are sent for all jobs.
Job Name	Yes	Name of the job.

Parameter	Mandatory	Description
Notification Type	Yes	Type of the notification. • When **Notification Scope** is **One job**, available options for this parameter include: – **Abnormal**: When a job is not running properly or fails, a notification is sent to notify the user of the abnormality. You can set **Max. Notifications** and **Min. Notification Interval (min)**. After a job encounters an exception or fails and before it is recovered, you can send the interval for sending alarm notifications. **NOTE** You can set **Max. Notifications** to a value from 1 to 50. If the default value **1** is used, **Min. Notification Interval (min)** is unavailable. You can set **Min. Notification Interval (min)** to a value from 5 to 60. – **Successful**: When a job runs successfully, a notification is sent to notify the user of the success. – **Uncompleted**: This function supports only the jobs scheduled by day. If the job execution time is later than the configured time by which the job has not finished, a notification is sent. – **Cancellation**: When a job is canceled, a notification is sent. **NOTE** An alarm notification is sent when a job being scheduled or a running job instance is manually stopped. If a user except the job executor cancels a job, a job cancellation alarm notification is sent. – **Successful rerun of a failed job** **NOTE** A notification will be sent after the successful rerun of a failed job only when a failure alarm was sent when the job failed. – Job modification A notification is sent when a job is modified or deleted, or the script used by the job is modified or deleted by a user except the job owner. If the job owner is empty, no alarm notification will be sent if the job is modified.

Parameter	Mandatory	Description
		– **Busy resources**: If the DLI resource queue is busy during job execution, the job execution takes a long time or fails. As a result, an alarm is generated and a notification is sent. • When **Notification Scope** is **All jobs**, available options for this parameter include: – **Abnormal**: When a job is not running properly or fails, a notification is sent to notify the user of the abnormality. You can set **Max. Notifications** and **Min. Notification Interval (min)**. After a job encounters an exception or fails and before it is recovered, you can send the interval for sending alarm notifications. **NOTE** You can set **Max. Notifications** to a value from 1 to 50. If the default value **1** is used, **Min. Notification Interval (min)** is unavailable. You can set **Min. Notification Interval (min)** to a value from 5 to 60. – **Cancellation**: When a job is canceled, a notification is sent. **NOTE** An alarm notification is sent when a job being scheduled or a running job instance is manually stopped. If a user except the job executor cancels a job, a job cancellation alarm notification is sent. – **Successful rerun of a failed job** **NOTE** A notification will be sent after the successful rerun of a failed job only when a failure alarm was sent when the job failed. – Job modification A notification is sent when a job is modified or deleted, or the script used by the job is modified or deleted by a user except the job owner. If the job owner is empty, no alarm notification will be sent if the job is modified. – **Busy resources**: If the DLI resource queue is busy during job execution, the job execution takes a long time or fails. As a result, an alarm is generated and a notification is sent.

Parameter	Mandatory	Description
		NOTE • For a real-time job, a notification is allowed to be sent only when the real-time job is in the **Run abnormally** or **Failed** state. For a batch job, a notification can be sent no matter when the batch job is in the **Run normally**, **Run abnormally**, or **Failed** state. • If you choose the default DLI resource queue, you may not be able to obtain the resources needed to perform operations because the queue is busy and other users may preempt resources. If this occurs, you may try again during off-peak hours or create a queue to run your workloads. • When a PatchData or test job is successfully executed, no notification is sent to avoid email or SMS bombing. In addition, no notification is sent when a PatchData job instance is recovered. • If a job is re-executed and succeeds after it fails, a job instance recovery notification is sent.
Notification Mode	Yes	• By topic • By owner
Topic Name	Yes	This parameter is mandatory only when **Notification Mode** is set to **By topic**. Select a notification topic. Click **View Topic** to go to the SMN page and view topics. **NOTE** Currently, only SMS, email, or HTTP are supported to subscribe to topics.
Terminal Protocol	Yes	Before setting this parameter, ensure that a **job alarm notification topic** has been configured in the workspace default configuration. This parameter is mandatory only when **Notification Mode** is set to **By owner**. • SMS • Email • Phone Click **Verify Contact Information** to check for the jobs for which no owner information is set. Click **View Subscription**. The **Terminal Subscriptions** page is displayed, on which you can view the terminal subscriptions that have been configured.

Parameter	Mandatory	Description
Cc	Yes	This parameter is mandatory only when **Notification Mode** is set to **By owner**. You can select up to 10 options.
Notification	Yes	Whether to enable the notification function. The function is enabled by default.

4. Click **OK**.

   NOTE

   - The DataArts Factory module sends notifications through SMN. Using SMN may incur fees. For pricing details, contact the SMN support personnel.
   - Multiple message topics can be configured for a job. When the job is successfully executed or fails to be executed, notifications can be sent to multiple subscribers.

Editing a Notification

After a notification is created, you can modify the notification parameters as required.

1. In the navigation pane on the DataArts Factory page, choose **Monitoring** > **Manage Notification**.
2. Click the **Notification Management** tab.
3. In the **Operation** column of a notification, click **Edit**. In the displayed dialog box, edit notification parameters. **Table 6-66** describes the notification parameters.
4. Click **Yes**.

Disabling a Notification

You can disable the notification function on the **Edit Notification** page or in the notification list.

1. In the navigation pane on the DataArts Factory page, choose **Monitoring** > **Manage Notification**.
2. Click the **Notification Management** tab.
3. In the **Notification Function** column, click  . When it changes to  , the notification function is disabled.

Viewing a Notification

You can view all notification information on the **Notification Records** tab page.

1. In the navigation pane on the DataArts Factory page, choose **Monitoring** > **Manage Notification**.
2. Click the **Notification Records** tab.

Deleting a Notification

If you no longer need a notification, perform the following operations to delete it:

1. In the navigation pane on the DataArts Factory page, choose **Monitoring** > **Manage Notification**.
2. Click the **Notification Management** tab.
3. You can delete a notification in either of the following ways:
   - In the **Operation** column of a notification, click **Delete**.
   - Select the notifications to delete and click **Batch Delete** above the notification list.
4. In the displayed dialog box, click **OK**.

6.7.6.2 Cycle Overview

Scenarios

Notifications can be set to specified personnel by day, week, or month, allowing related personnel to regularly understand job scheduling information about the quantity of successfully/unsuccessfully scheduled jobs and failure details.

Constraints

This function depends on OBS.

Prerequisites

- Simple Message Notification (SMN) has been enabled, topics have been configured, and subscriptions have been added to the topics.
- Jobs are not in **Not started** status and have been submitted.
- OBS has been enabled and a folder has been created in OBS.

Creating a Notification

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the navigation pane on the DataArts Factory page, choose **Monitoring** > **Manage Notification**.
3. On the **Cycles** tab page, click **Create Notification**. In the displayed dialog box, configure parameters. **Table 6-67** describes the notification parameters.

Figure 6-99 Create a notification

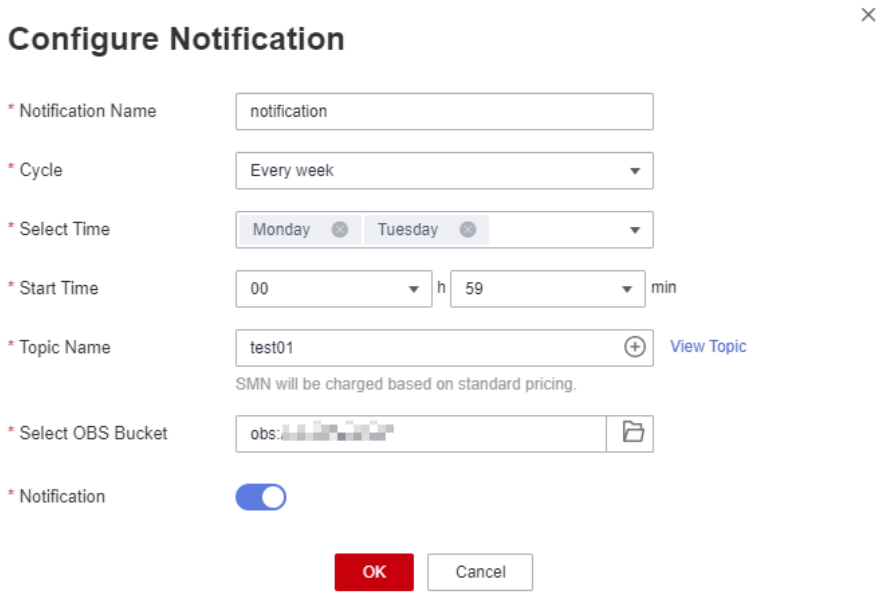


Table 6-67 Notification parameters

Parameter	Mandatory	Description
Notification Name	Yes	Name of the notification to be sent.
Cycle	Yes	Interval for sending notifications, which can be set to **Daily**, **Weekly**, or **Monthly**. **NOTE** When **Cycle** is set to **Daily**, **Weekly**, or **Monthly**, a notification is sent every day, week, or month, and the notification content comes from the data generated from the last 24 hours, seven days, or 30 days.
Select Time	Yes	This parameter is mandatory when **Cycle** is set to **Weekly** or **Monthly**. Time when the notification is sent. • If **Cycle** is set to **Weekly**, the value can be any day or any several days from Monday to Sunday in a week. • If **Cycle** is set to **Monthly**, the value can be any day or any several days from 1st to 31st in a month.
Start Time	Yes	Point in time when the notification is sent. The value can be accurate to hour or minute.
Topic Name	Yes	Notification topic

Parameter	Mandatory	Description
OBS Bucket	Yes	OBS bucket for storing notification records
Notification	Yes	Whether to enable the notification function. The function is enabled by default.

4. Click **OK**.

   NOTE

   DataArts Factory sends notifications through SMN. Using SMN may incur fees. For pricing details, contact the SMN support personnel.

5. After the notification is created, you can perform the following operations on the notification:
   - Click **Edit**. In the **Create Notification** dialog box, edit the notification again.
   - Click **View Record**. In the **View Record** dialog box, view the job scheduling details.
   - Click **Delete**. In the **Delete Notification** dialog box, click **OK** to delete the notification.

6.7.6.3 Managing Terminal Subscriptions

Scenario

You can configure terminal subscriptions (SMS messages, emails, and phone calls) by owner. After configuring a subscription, you can use the Manage Notification function to configure a job notification task. When a job runs abnormally or successfully, notifications are sent to the configured owners.

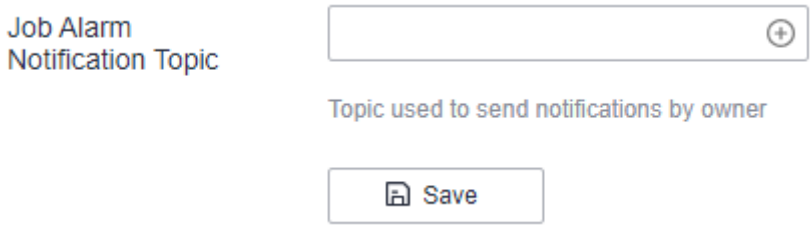
Prerequisites

Message notification has been enabled and a topic has been configured. Before configuring subscriptions by owner, ensure that you have set a **job alarm notification topic** for the workspace.

Creating a Notification

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the left navigation pane on the DataArts Factory console, choose **Configuration** > **Configure**. Choose **Default Configuration**. For details about how to configure alarm notification topics for workspace jobs by owner, see **Job Alarm Notification Topic**. If you have configured an alarm notification topic, skip this step.

Figure 6-100 Setting Job Alarm Notification Topic



3. In the navigation pane on the DataArts Factory page, choose **Monitoring** > **Manage Notification**.
4. Click the **Terminal Subscriptions** tab and click **Add Subscription**. In the displayed dialog box, set required parameters.

Figure 6-101 Adding a subscription

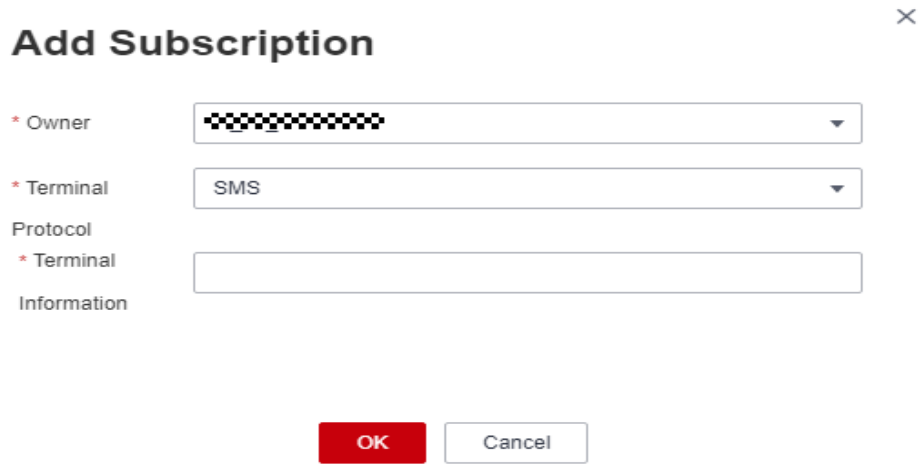


Table 6-68 Parameters for adding a subscription

Parameter	Mandatory	Description
Owner	Yes	Set the subscription owner, which was configured during job creation.
Terminal Protocol	Yes	• SMS • Email • Phone
Terminal information	Yes	Set the information about the terminal.

5. Click **OK**.
6. After the terminal subscription is created, you can perform the following operations on the notification:

- Click **Request Subscription**. In the displayed dialog box, the subscription status is **Unconfirmed**. After you click **OK**, the subscription status becomes **Confirmed**.
- Click **Delete**. In the **Delete Subscription** dialog box, click **OK** to delete the subscription.

  NOTE

  You can request or delete subscriptions, but cannot edit them.

7. After the preceding operations are complete, configure job alarm notifications by owner on the **Managing Notifications** page.

6.7.7 Managing Backups

You can back up all jobs, scripts, resources, and environment variables at a specified interval.

You can also restore assets that have been backed up, including jobs, scripts, resources, and environment variables.

Constraints

This function depends on OBS.

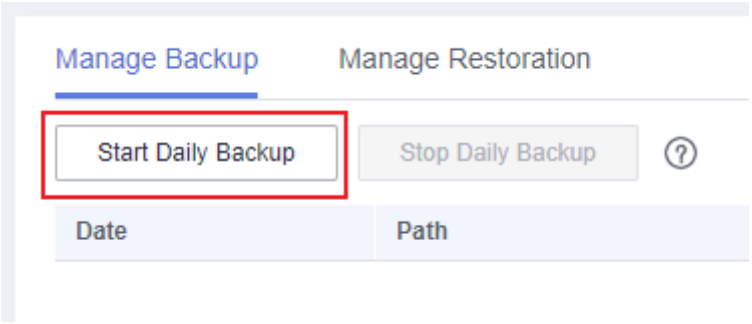
Prerequisites

OBS has been enabled and a folder has been created in OBS.

Backing Up Assets

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the navigation tree on the left, choose **Manage Backup**.
3. Click **Start Daily Backup**. In the **Browse OBS File** dialog box, select an OBS folder.

Figure 6-102 Managing backup



NOTE

- Daily Backup starts at 00:00 every day to back up all jobs, scripts, resources, and environment variables of the previous day. The jobs, scripts, resources, and environment variables of the previous day are not backed up on the current day.
- If you select only the bucket name as the OBS storage path, the backup object is automatically stored in the folder named after the backup date. Environment variables, resources, scripts, and jobs are stored in the **1_env**, **2_resources**, **3_scripts**, and **4_jobs** folders, respectively.
- After the backup is successful, the **backup.json** file is automatically generated in the folder named after the backup date. The file stores job information based on the node type and can be modified before job restoration.
- To stop daily backup, click **Stop Daily Backup**.

Restoring Assets

Step 1 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 2 In the navigation tree of the DataArts Factory console, choose **Manage Backup**.

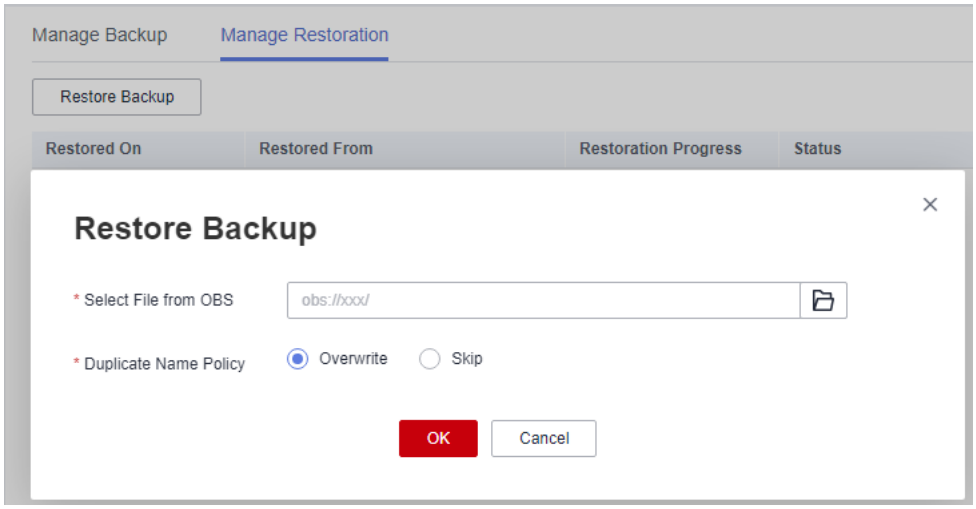
Step 3 On the **Manage Restoration** tab, click **Restore Backup**.

In the **Restore Backup** dialog box, select the storage path of the asset to be restored from the OBS bucket and set the duplicate name policy.

NOTE

- The storage path is the file path generated in **Backing Up Assets**.
- Before restoring assets, you can modify the **backup.json** file in the backup path. You can change the connection name (connectionName), database name (database), and cluster name (clusterName).

Figure 6-103 Restoring assets



Step 4 Click **OK**.

----End

6.7.8 Operation History

You can view historical operations on the **Operation History** page. The system stores data for a maximum of three months and automatically deletes older data.

1. Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.
2. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
3. In the left navigation pane of the DataArts Factory console, choose **Monitoring** > **Operation History**.
4. You can perform the following operations on this page:
   - Filter out historical operations in a specified time period.
   - Filter out historical operations related to job names or node names by involved object.
   - Perform a fuzzy search of historical operations.
   - Filter out historical operations by operation object, operation type, operators, or status.

6.8 Configuration and Management

6.8.1 Configuring Resources

6.8.1.1 Configuring Environment Variables

This topic describes how to configure and use environment variables.

Application Scenario

Configure job parameters. If a parameter belongs to multiple jobs, you can extract this parameter as an environment variable. Environment variables can be imported and exported.

NOTE

The roles that can configure workspace environment variables in the simple and enterprise mode are as follows:

Simple mode: Both developers and administrators can create and edit environment variables in a workspace. This mode does not distinguish the development environment from the production environment. Developers can modify environment variables.

Enterprise mode: Only administrators can create or edit environment variables in a workspace.

Importing Environment Variables

This function is available only if the OBS service is available. If OBS is unavailable, variables can be imported from the local PC.

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

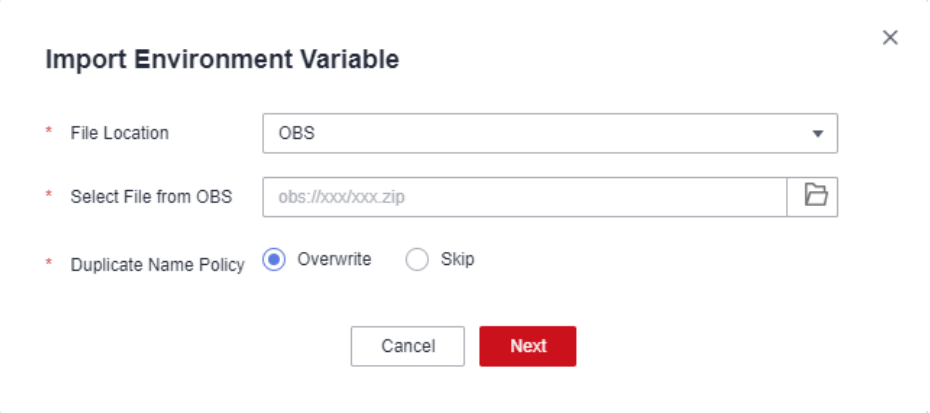
Step 2 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 3 In the navigation tree on the left, choose **Specifications**.

Step 4 Click **Environment Variables**. On the **Environment Variables** page, click **Import**.

Step 5 In the **Import Environment Variable** dialog box, select an environment variable file from OBS or a local path and the duplicate name policy.

Figure 6-104 Importing Environment Variables



----End

Configuration Method

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

Step 2 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 3 In the navigation tree on the left, choose **Specifications**.

Step 4 On the **Environment Variable** page, set the variables or constants listed in **Table 6-69** and click **Save**.

NOTE

The difference between a variable and a constant lies in whether their values need to be reconfigured when they are imported to another workspace or project.

- The value of a variable (such as **workspace name**) varies depending on the workspace. When exporting a variable from a workspace and import it to another workspace, you must reconfigure its value.
- The value of a constant in different workspaces is the same. When importing a constant to another workspace, you do not need to reconfigure its value.

Figure 6-105 Configuring environment variables

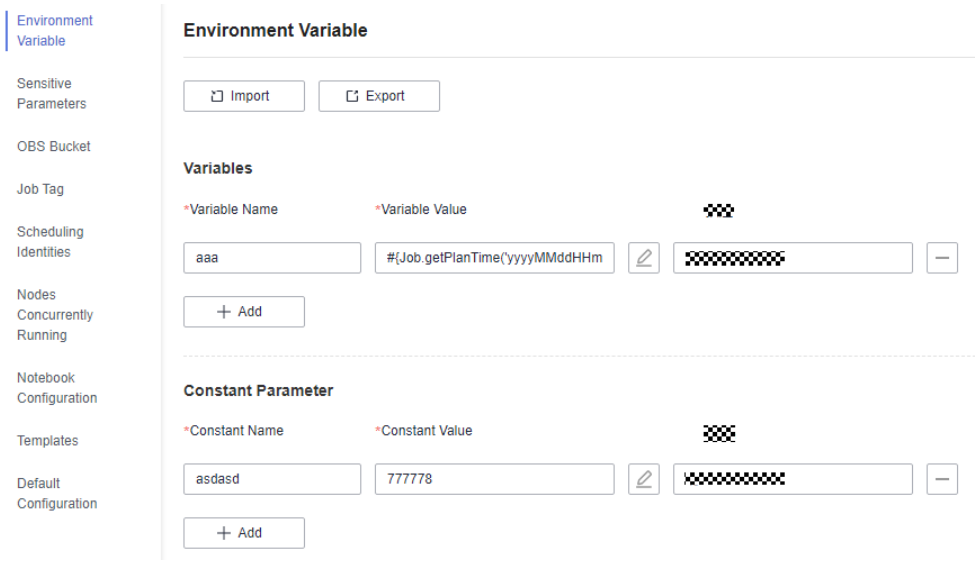


Table 6-69 Configuring environment variables

Parameter	Mandatory	Description
Parameter	Yes	The parameter name must be unique, consist of 1 to 64 characters, and contain only letters, digits, underscores (_), and hyphens (-). The parameter name must be in the format set in **Configuring Script Variables**. For example, if the format set in the script variable definition is **${dlf.}**, the parameter name must be set to **dlf.***xxx*.
Value	Yes	Parameter values support constants and EL expressions but do not support system functions. For example, **123** and **abc** are supported. If the parameter value is a string, add double quotation marks (""), for example, **"05"**. For details about how to use EL expressions, see **Expression Overview**.
Description	No	Parameter description

After configuring an environment variable, you can add, edit, or delete it.

- **Add**: Click **Add** to add an environment variable.
- **Edit**: If the parameter value is a constant, change the parameter value in the text box. If the parameter value is an EL expression, click  next to the text box to edit the EL expression. Click **Save**.

- **Delete**: Click next to the parameter value text box to delete the environment variable.

----End

How-Tos

The configured environment variables can be used in either of the following ways:

1. ${Environment variable}
2. #{Evn.get("Environment variable")}

Example

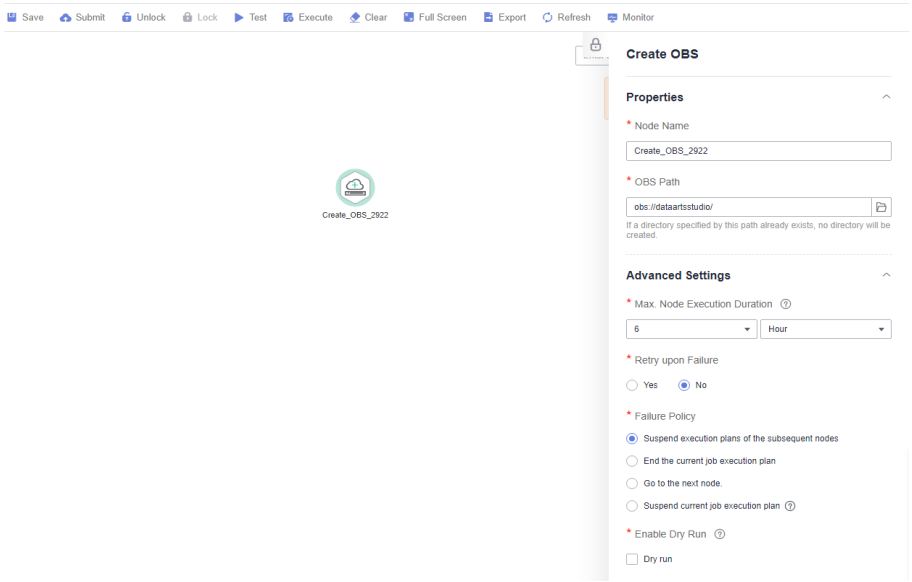
Context:

- A job named **test** has been created in the DataArts Factory module.
- An environment variable has been added. The parameter name is **job** and the parameter value is **123**.

Step 1 Open **test** and drag a **Create OBS** node from the node library.

Step 2 On the **Node Properties** tab page, configure the node properties.

Figure 6-106 Configuring parameters for the Create OBS node



Step 3 Click **Save** and then **Monitor** to monitor the running status of the job.

----End

6.8.1.2 Configuring an OBS Bucket

The execution history of scripts, jobs, and nodes is stored in OBS buckets. If no OBS bucket is available, you cannot view the execution history. This section describes how to configure an OBS bucket.

Constraints

The OBS path is only supported for OBS buckets and not for parallel file systems.

Procedure

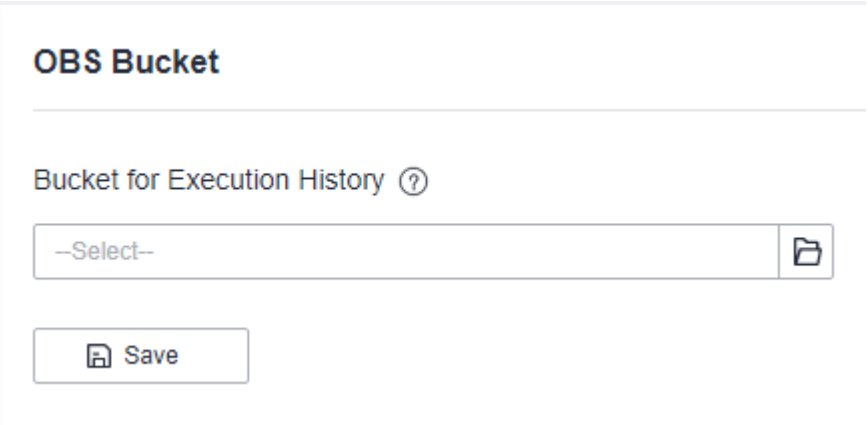
Step 1 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 2 In the navigation pane, choose **Configuration** > **Configure**.

Step 3 Choose **OBS Bucket**.

Step 4 Select an OBS bucket.

Figure 6-107 Configuring an OBS bucket



Step 5 Click **Save**.

----End

6.8.1.3 Managing Job Tags

Job tags are used to label jobs of the same or similar purposes to facilitate job management and query. This section describes how to manage job tags, including adding, deleting, importing, and exporting tags.

Adding a Job Tag

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

Step 2 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 3 In the navigation pane, choose **Configuration** > **Configure**.

Step 4 Choose **Job Tag**.

Step 5 Click **Add**. In the displayed dialog box, enter a tag name and click **OK**.

NOTE

You can add a maximum of 100 job tags.

----End

Deleting a Job Tag

Step 1 In the navigation pane, choose **Configuration** > **Configure**.

Step 2 Choose **Job Tag**.

Step 3 Locate the tag you want to delete and click **Delete** in the **Operation** column. In the displayed dialog box, click **OK**.

NOTE

A locked tag cannot be deleted. For details about how to unlock a tag, see **Locking and Unlocking a Job Tag**.

----End

Monitoring Jobs with a Specified Tag

Step 1 In the navigation pane, choose **Configuration** > **Configure**.

Step 2 Choose **Job Tag**.

Step 3 Locate a tag and click **Monitor** in the **Operation** column. The **Monitor Job** page is displayed, on which all the jobs with the tag are displayed.

----End

Locking and Unlocking a Job Tag

To perform these operations, you must have the **DAYU Administrator**, **Tenant Administrator**, or workspace administrator permission.

Step 1 In the navigation pane, choose **Configuration** > **Configure**.

Step 2 Choose **Job Tag**.

Step 3 Locate a tag and click **Lock** or **Unlock** in the **Operation** column.

NOTE

- A locked job tag cannot be deleted.
- Importing a locked tag will fail.
- A locked tag cannot be added to or removed from a job.
- Importing a job with a locked tag will fail.
- When a job fails to be imported and a tag needs to be automatically generated, if the tag already exists and is locked, it will not be added to the job.

----End

Importing Job Tags

To perform this operation, you must have the **DAYU Administrator**, **Tenant Administrator**, or workspace administrator permission.

Step 1 In the navigation pane, choose **Configuration** > **Configure**.

Step 2 Choose **Job Tag**.

Step 3 Click **Import Job Tag**.

Step 4 In the displayed dialog box, set the following parameters:

- **File Location**: Select **Local** or **OBS**.
- **Select File from Local/OBS**: Select a local path or an OBS bucket path.

  NOTE

  - You are advised to obtain a file to import by exporting tags. The first row of the file is the tag name, and the first column is the job name. If a job has a specified tag, the value in the corresponding cell is 1. Otherwise, the value is 0. If a cell is empty, the system uses value 0 for the cell.
  - The maximum size of the file to be imported is 10 MB.
  - If the file to be imported contains two tags with the same name, and the ID of one tag is 0 and that of the other tag is 1, the system uses 1 as the tag ID.
  - If the file to be imported contains two jobs with the same name, the system identifies the job in the latter row and uses the tag ID in this row.
- **Mode**: Select **Append** or **Overwrite**.
  - **Append**: The new tag will not overwrite the existing one.
  - **Overwrite**: The new tag will overwrite the existing one.

Step 5 Click **OK**.

----End

Exporting Job Tags

Step 1 In the navigation pane, choose **Configuration** > **Configure**.

Step 2 Choose **Job Tag**.

Step 3 Export tags.

- To export all tags, click **Export All Tags** above the tag list.
- To export some tags, select them and click **Export Selected Tags** above the tag list.

The following figure shows the exported job tags.

Figure 6-108 Exporting job tags

	A	B	C	D	E	F
1	jobName	DWS_TRANSFORM	Invalid clust	The MRS cluster na	The cluster associated w	The cluster associat
2	job_foreach	1	0	1	0	1
3	job_real	0	0	1	1	0
4	job_weituo	0	1	0	0	1
5	job_subjob	1	1	0	1	0
6	job_ETL_dli2dws	0	0	1	1	1
7	job_foreach_copy	1	0	0	1	1
8	job_ETL_copy	0	0	1	0	0
9	guowangTest	1	1	0	0	1
10	guowangTest_qjxtest	1	0	0	0	0
11	qjxForeach	1	0	1	1	1
12	job_timlx_1	0	1	1	0	0
13	job_timlx_2	0	0	0	0	1
14	guowangTest_copy_hdfs2hiv	0	0	1	0	0

NOTE

- In the exported file, the first row is the tag name, and the first column is the job name. If a job has a specified tag, the value in the corresponding cell is 1. Otherwise, the value is 0.
- The first column displays names of all the jobs in the workspace, including real-time job nodes, For Each subjobs, and Subjob subjobs.

----End

6.8.1.4 Configuring a Scheduling Identity

The following problems may occur during job execution in DataArts Factory:

- The job execution mechanism of the DataArts Factory module is to execute the job as the user who starts the job. For a job that is executed in periodic scheduling mode, if the IAM account used to start the job is suspended or deleted during the scheduling period, the system cannot obtain the user identity authentication information. As a result, the job fails to be executed.
- If a job is started by a low-privilege user, the job fails to be executed due to insufficient permissions.

To address these issues, you can configure an identity for scheduling jobs. During job scheduling, this identity interacts with other services, preventing the above job execution failures.

NOTE

During the periodic scheduling of a job, if the default user of the job is deleted and another user submits a version and schedules the job, the user who submits the version is considered as the executor of the job by default.

Classification of Scheduling Identities

Scheduling identities are classified into agencies and IAM accounts.

- Agencies: Cloud services interwork with each other, and some cloud services are dependent on other services. You can create an agency to delegate cloud services to access other services and perform resource O&M on your behalf.

  Agencies are classified into the following types:

  - Public agencies: They apply to all jobs in the workspace. For details about how to configure a public agency, see **Configuring a Public Agency**.
  - Job agencies: They apply only to a single job. For details about how to configure a job agency, see **Configuring a Job-Level Agency**.
- IAM accounts: You can configure IAM accounts through user groups in a unified manner and manage permissions in an easier way than agencies. IAM accounts also have better compatibility and support MRS nodes (MRS Presto SQL, MRS Spark, MRS Spark Python, MRS Flink Job, and MRS MapReduce), directly connected nodes (MRS Spark SQL and MRS Hive SQL), and ETL Job nodes whose destination is DWS, so IAM accounts can be used to submit jobs for some MRS clusters and ETL Job nodes that cannot be submitted through agencies.

  IAM accounts are classified into the following types:

- Public IAM accounts: They apply to all jobs in the workspace. For details about how to configure a public IAM account, see **Configuring a Public IAM Account**.
- Execution users: They apply only to a single job. For details about how to configure an execution user, see **Configuring an Executor**.

Priorities of Scheduling Identities

The system obtains permissions for the job agency, public agency, execution user, and public IAM account in sequence, and then executes jobs with the permissions.

By default, a job is executed by the user who starts the job. If a job is started by a user without the required permissions, the job fails to be executed due to insufficient permissions. You can configure a scheduling identity to resolve this issue.

Constraints

- To create or modify an agency, you must have the **Security Administrator** permissions.
- To configure a workspace-level scheduling identity, you must have the **DAYU Administrator** or **Tenant Administrator** policy.
- To configure a job-level agency, you must have the permission to view the list of agencies.

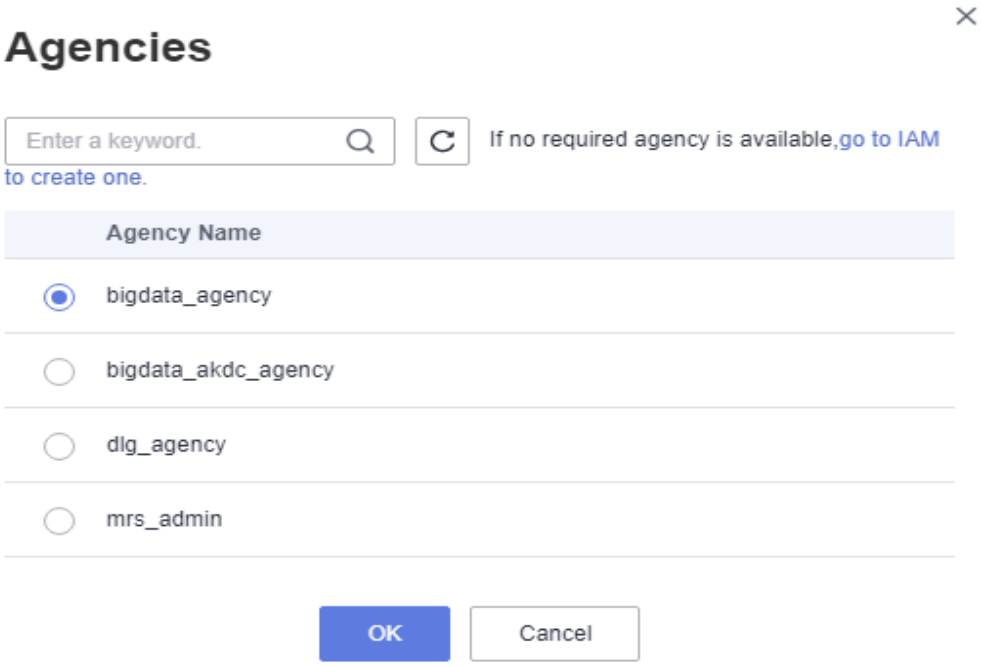
Configuring a Public Agency

CAUTION

A public agency applies to all jobs in the workspace, especially those that contain MRS nodes. Exercise caution when performing this operation.

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the navigation pane, choose **Configuration** > **Configure**.
3. Choose **Scheduling Identities** and set **Public Scheduling Identity** to **Public agency**.
4. Click **+** to select an agency or create one. For how to create an agency and configure permissions, see **Reference: Creating an Agency** and **Reference: Configuring Agency Permissions**.

Figure 6-109 Configuring a workspace-level agency



5. Click **OK** to return to the **Scheduling Identities** page and click .

   NOTE

   For a batch processing job, a public agency takes effect in the next cycle. For a real-time processing job, you must restart the job for a public agency to take effect.

Configuring a Job-Level Agency

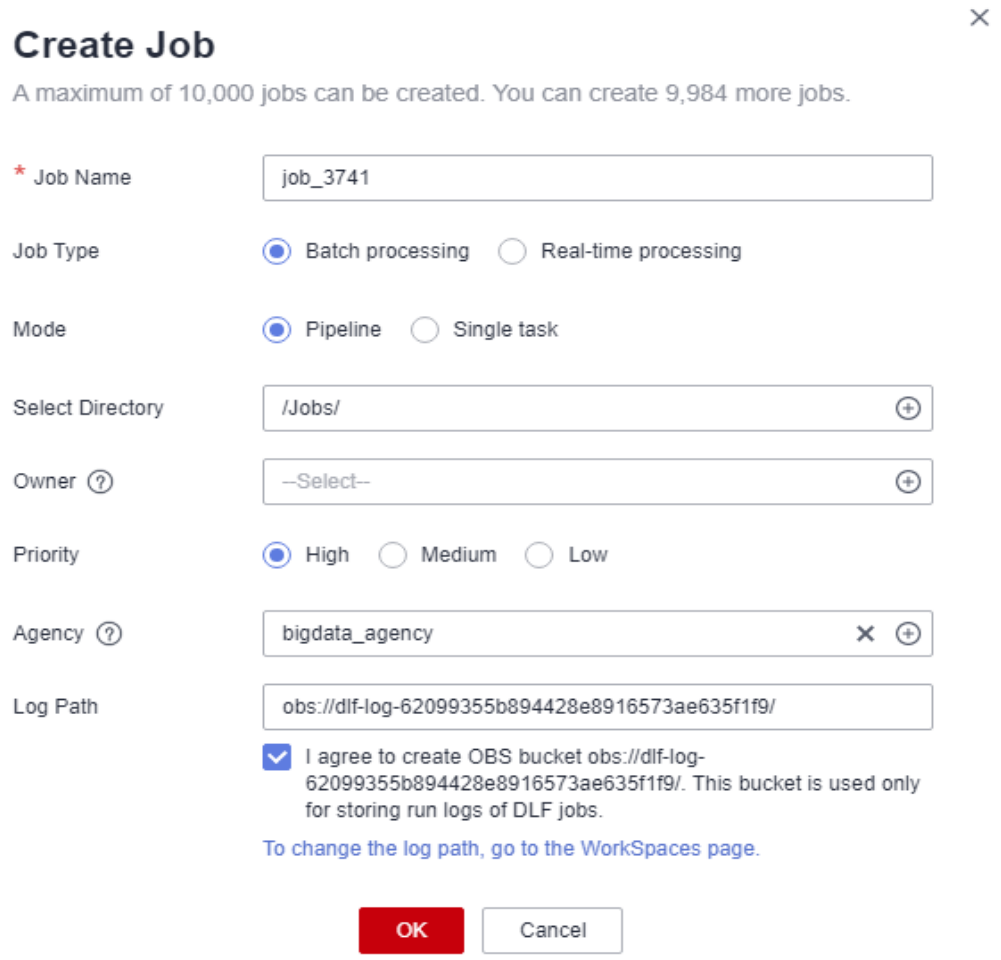
NOTE

You can create a job-level agency when creating a job. You can also modify the agency of an existing job.

Configuring an agency when creating a job

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the navigation pane of the DataArts Factory homepage, choose **Development** > **Develop Job**.
3. Right-click the job directory and choose **Create Job** from the shortcut menu. The **Create Job** dialog box is displayed. If a workspace-level agency has been configured, it is used for the job by default. You can also select another agency from the agency list. For how to create an agency and configure permissions, see **Reference: Creating an Agency** and **Reference: Configuring Agency Permissions**.

Figure 6-110 Configuring an agency for a job



Modifying the agency of an existing job

1. In the navigation pane of the DataArts Factory homepage, choose **Development** > **Develop Job**.
2. In the job directory, double-click an existing job. On the far right of the displayed page, click **Basic Info**. The dialog box of the job's basic settings is displayed. If a workspace-level agency has been configured, it is used by default. You can also select another agency from the agency list.

Configuring a Public IAM Account

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the navigation pane, choose **Configuration** > **Configure**.
3. Choose **Scheduling Identities** and set **Public Scheduling Identity** to **Public IAM account**.
4. Enter the public IAM account in the text box.
5. Click .

Configuring an Executor

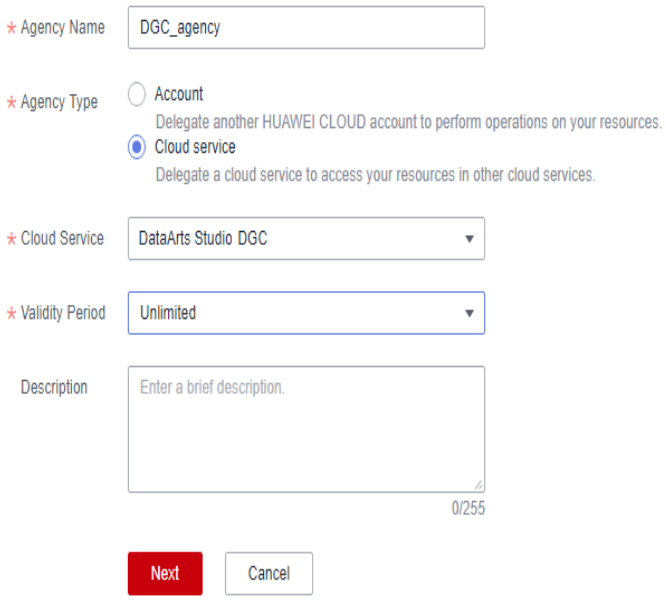
Configuring a Job Executor

1. In the job directory, double-click a job.
2. Click the **Basic Info** tab and set the executor for the job.

Reference: Creating an Agency

1. Log in to the IAM console.
2. In the navigation pane, choose **Agencies** and click **Create Agency**.
3. Enter an agency name, for example, **DGC_agency**.
4. On the displayed page, select **Cloud service** for **Agency Type** and **Data Lake Governance Center (DGC)** for **Cloud Service**. This grants operation permissions to DataArts Studio so that DataArts Studio can use cloud services and perform O&M for you.

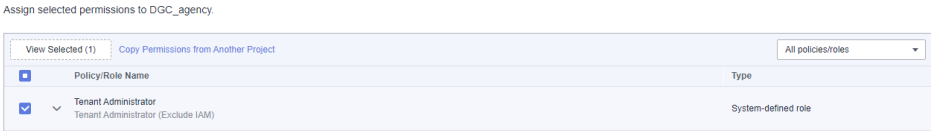
   Figure 6-111 Creating an agency



5. Click **Next**.
6. On the **Authorize Agency** page, search for and select the **Tenant Administrator** policy. Then click **Next**.
   - Users assigned the **Tenant Administrator** policy have all permissions on all services except on IAMIAM. Therefore, delegate the **Tenant Administrator** policy to DataArts Studio so that DataArts Studio can access all related services.
   - If you want to meet the security control requirements for fewer permissions, you only need to configure the **OBS OperateAccess** permissions (During job execution, execution log information needs to be written to OBS. Therefore, you need to add the **OBS OperateAccess** permissions.) Then, configure different agency permissions based on the

node type in the job. For example, if a job contains only the **Import GES** node, you can configure the **GES Administrator** and **OBS OperateAccess** permissions. For details, see **Reference: Configuring Agency Permissions**.

Figure 6-112 Assigning permissions



7. Click **OK**.

Reference: Configuring Agency Permissions

After the operation permissions of an account are delegated to DataArts Studio, you must configure the permissions of the agency identity so that DataArts Studio can interact with other services.

For purposes of permissions minimization, you can configure the **Admin** permissions for services based on the node types in jobs. For details, see **Table 6-70**.

The **Admin** permissions can also be configured based on the operations, resources, and request conditions for a specific service. Based on the node types in jobs, permissions are defined by service APIs to allow for more fine-grained, secure access control of cloud resources. Configure the permissions according to **Table 6-71**. For example, for a job containing the **Import GES** node, you only need to create a custom policy and select **ges:graph:getDetail** (viewing graph details), **ges:jobs:getDetail** (querying task status), and **ges:graph:access** (using graphs).

NOTICE

- An MRS cluster supports job submission through an agency if either of the following conditions is met:
  - It is a non-security cluster.
  - It is a security cluster whose version is later than 2.1.0 and which has MRS 2.1.0.1 or later installed.
- If an MRS cluster does not support job submission through an agency, agencies cannot be configured for the jobs that contain the following nodes:

  MRS-related nodes (MRS Presto SQL, MRS Spark, MRS Spark Python, MRS Flink Job, and MRS MapReduce) and MRS Spark SQL and MRS Hive SQL nodes connected through APIs.

- Configure the service-level **Admin** permissions.

  During job execution, execution log information needs to be written to OBS. Therefore, the **OBS OperateAccess** permissions must be added for all jobs during coarse-grained authorization.

Table 6-70 The admin permissions for related nodes

Node Name	System Permission	Description
CDM Job, DIS Stream, DIS Dump, and DIS Client	DAYU Administrator	All DataArts Studio permissions
Import GES	GES Administrator	Permissions required to perform all operations on GES. This role depends on the **Tenant Guest** and **Server Administrator** roles in the same project.
• MRS Presto SQL, MRS Spark, MRS Spark Python, MRS Flink Job, and MRS MapReduce • MRS Spark SQL and MRS Hive SQL (connecting to MRS clusters through MRS APIs)	MRS Administrator MRS Fullaccess KMS Administrator	MRS Administrator: all execute permissions of MRS specified in the RBAC policy This role depends on the **Tenant Guest** and **Server Administrator** roles in the same project. MRS Fullaccess: MRS administrator permission specified in the fine-grained policy Users assigned the **KMS Administrator** role have the administrator permissions for encryption keys in DEW.
MRS Spark SQL, MRS Hive SQL, MRS Kafka, and Kafka Client (connecting to the clusters in proxy mode)	DAYU Administrator KMS Administrator	**DAYU Administrator** has all permissions required for DataArts Studio. Users assigned the **KMS Administrator** policy have the administrator permissions for encryption keys in DEW.
DLI Flink Job, DLI SQL, and DLI Spark	DLI Service Admin	All operation permissions for DLI.
DWS SQL, RDS SQL (connecting to data sources in proxy mode), and Shell	DAYU Administrator KMS Administrator	**DAYU Administrator** has all permissions required for DataArts Studio. Users assigned the **KMS Administrator** policy have the administrator permissions for encryption keys in DEW.

Node Name	System Permission	Description
CSS	DAYU Administrator Elasticsearch Administrator	**DAYU Administrator** has all permissions required for DataArts Studio. Users assigned the **Elasticsearch Administrator** policy have all permissions for CSS. This role depends on the **Tenant Guest** and **Server Administrator** roles in the same project.
Create OBS, Delete OBS, and OBS Manager	OBS OperateAccess	Basic object operation permissions, such as viewing buckets, uploading objects, obtaining objects, deleting objects, and obtaining object ACLs.
SMN	SMN Administrator	All operation permissions for SMN.

- Configure fine-grained permissions. (Create custom policies based on the actions supported by each service.)

  For details on how to create a custom policy, see **Creating a Custom Policy**.

  NOTE

  - During job execution, you must write execution logs to OBS. When the fine-grained authorization mode is used, the following OBS permissions need to be added for all types of jobs:
    - obs:bucket:GetBucketLocation
    - obs:object:GetObject
    - obs:bucket:CreateBucket
    - obs:object:PutObject
    - obs:bucket:ListAllMyBuckets
    - obs:bucket:ListBucket
  - CDM Job, DIS Stream, DIS Dump and DIS Client nodes belong to the DataArts Studio module. DataArts Studio does not support fine-grained authorization. Therefore, only the **DataArts Studio Administrator** policy can be configured for jobs containing these types of nodes.
  - CSS does not support fine-grained authorization and requires a proxy. Therefore, the **DataArts Studio Administrator** and **Elasticsearch Administrator** policies can be configured for jobs containing these nodes.
  - SMN does not support fine-grained authorization. Therefore, jobs containing these nodes require the **SMN Administrator** permissions.

Table 6-71 Creating a custom policy

Node Name	Action
Import GES	• ges:graph:access • ges:graph:getDetail • ges:jobs:getDetail
• MRS Presto SQL, MRS Spark, MRS Spark Python, MRS Flink Job, and MRS MapReduce • MRS Spark SQL and MRS Hive SQL (connecting to MRS clusters through MRS APIs)	• mrs:job:delete • mrs:job:stop • mrs:job:submit • mrs:cluster:get • mrs:cluster:list • mrs:job:get • mrs:job:list • kms:dek:crypto • kms:cmk:get
MRS Spark SQL, MRS Hive SQL, MRS Kafka, and Kafka Client (connecting to the clusters in proxy mode)	• kms:dek:crypto • kms:cmk:get • DataArts Studio Administrator (role)
DLI Flink Job, DLI SQL, and DLI Spark	• dli:jobs:get • dli:jobs:update • dli:jobs:create • dli:queue:submit_job • dli:jobs:list • dli:jobs:list_all
DWS SQL, RDS SQL (connecting to data sources in proxy mode), and Shell	• kms:dek:crypto • kms:cmk:get • DataArts Studio Administrator (role)
Create OBS, Delete OBS, and OBS Manager	• obs:bucket:GetBucketLocation • obs:bucket:ListBucketVersions • obs:object:GetObject • obs:bucket:CreateBucket • obs:bucket:DeleteBucket • obs:object:DeleteObject • obs:object:PutObject • obs:bucket:ListAllMyBuckets • obs:bucket:ListBucket

6.8.1.5 Configuring the Number of Concurrently Running Nodes

This section describes how to configure the maximum number of job nodes that can run concurrently in a workspace.

Constraints

The number of concurrently running nodes in the workspace cannot exceed that in the instance.

Procedure

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

Step 2 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 3 In the navigation pane, choose **Configuration** > **Configure**.

Step 4 Choose **Nodes Concurrently Running**.

Step 5 Set **Nodes Concurrently Running in the Workspace**. Ensure that the value is less than or equal to the maximum number of nodes that can run concurrently in the DataArts Studio instance.

Table 6-72 lists the maximum number of nodes that can run concurrently in the DataArts Studio instance. To view the quota of the job node scheduling times per day, click **More** of a DataArts Studio instance and select **Quota Usage**.

Table 6-72 Maximum number of nodes that can run concurrently in a DataArts Studio instance

Job Node Scheduling Times/Day of a DataArts Studio Instance	Maximum Number of Nodes That Can Run Concurrently in a DataArts Studio Instance
<=500	10
<=5000	50
<=20000	100
<=40000	200
<=80000	300
> 80000	400

Figure 6-113 Configuring the number of concurrently running nodes

Step 6 Click **Save**.

----End

Viewing the Number of Historical Nodes Concurrently Running

Step 1 In the navigation pane, choose **Configuration** > **Configure**.

Step 2 Choose **Nodes Concurrently Running**.

Step 3 In the **Historical Nodes Concurrently Running** area, set the time range.

Step 4 Click **OK**.

NOTE

The maximum time range is 24 hours.

----End

6.8.1.6 Configuring a Template

This section describes how to create and use a Flink SQL template. When writing Flink SQL code, you can use an SQL template for repeated service logic. In addition, you can use a job parameter template when configuring job parameters.

Constraints

This function applies to the following scenarios:

- Use a script template for a Flink SQL script.
- During pipeline job development, use a Fink SQL script which uses a script template for the MRS Flink Job node and use a parameter template for **Program Parameter** of the MRS Flink Job node.
- Use a script template in a single-task Flink SQL job.
- Use template parameters in a single-task Flink JAR job.

Procedure

Step 1 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 2 In the navigation pane, choose **Configuration** > **Configure**.

Step 3 Choose **Templates**.

- Create a script template.
  a. On the **Script Templates** page, click **Add**.
  b. Set **Template Name**.
  c. Enter an SQL statement and reference script parameters.
  d. Configure the script template parameters. The parameter names cannot be changed, and the parameter values can be changed.

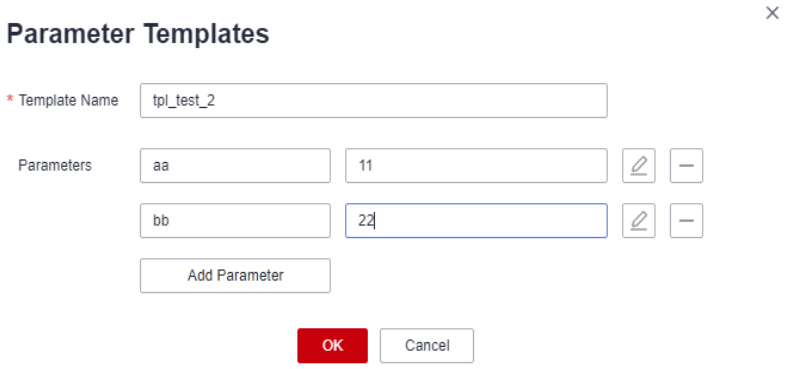
  Figure 6-114 Creating a script template

  e. Click **Save**.

  You can view, modify, or delete the created script template.
- Create a parameter template.
  a. On the **Parameter Templates** page, click **Add**.
  b. Set **Template Name**.
  c. Click **Add Parameter** and set parameter names and values. You can modify or delete parameters.

  Figure 6-115 Creating a parameter template

  d. Click **OK**.

  You can view, modify, or delete the created parameter template.

For details about the application scenarios of script templates and parameter templates, see **Using Script Templates and Parameter Templates**.

----End

6.8.1.7 Configuring a Scheduling Calendar

You can configure a scheduling calendar and specify the working days for scheduling a job.

Constraints

This function is available for batch processing jobs but not for real-time processing jobs.

Procedure

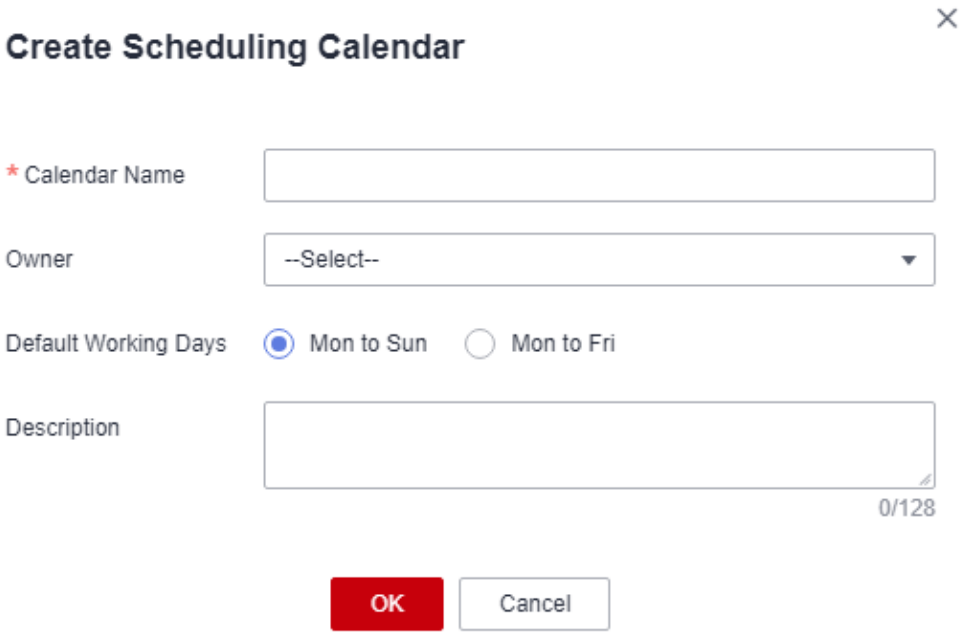
Step 1 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 2 In the left navigation pane, choose **Configuration** > **Configure**.

Step 3 Choose **Scheduling Calendars**.

Step 4 Click **Add**. The **Create Scheduling Calendar** dialog box is displayed.

Figure 6-116 Create Scheduling Calendar dialog box



Step 5 Set parameters for the scheduling calendar.

Set **Calendar Name**, **Owner**, **Default Working Days**, and **Description**.

You can select **Mon to Fri** or **Mon to Sun** for **Default Working Days**. By default, **Mon to Fri** is selected.

Step 6 Click **OK**.

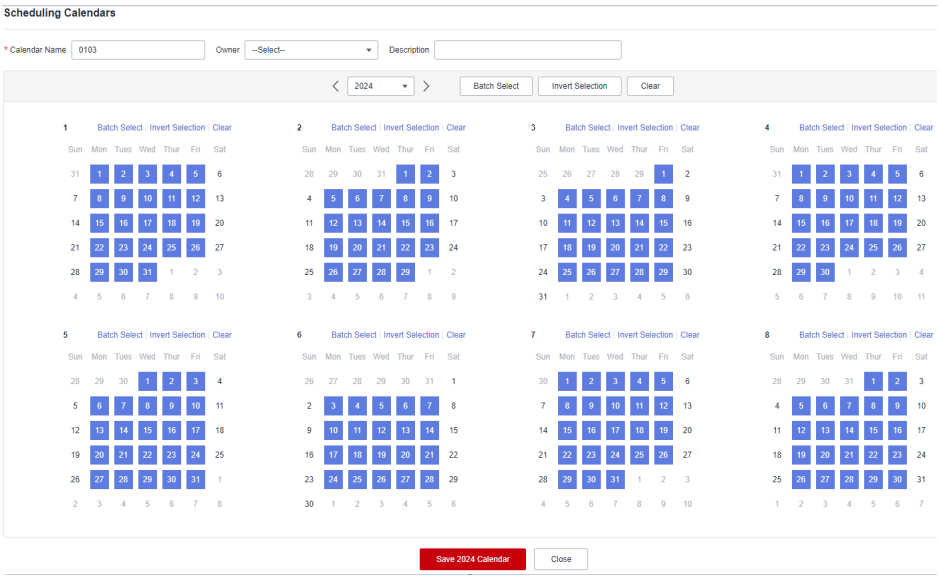
After creating the calendar, you can use it for jobs. Go to the DataArts Factory console, open a job, click **Scheduling Setup**, select **Run periodically** for **Scheduling Type**, and select the calendar you have created for **Scheduling Calendar**.

----End

More Operations

- Modify a calendar: Click **Modify** in the **Operation** column.
  - **Batch Select**: Select all Mondays to Fridays of the current month.
  - **Invert Selection**: Select all non-working days.
  - **Clear**: Clear selected working days.

  Figure 6-117 Modifying a scheduling calendar



- Delete a calendar: Click **Delete** in the **Operation** column.

6.8.1.8 Configuring a Default Item

This section describes how to configure a default item. You can perform the operations in this section only if you have the permissions of **DAYU Administrator** or **Tenant Administrator**.

Scenario

If a parameter is invoked by multiple jobs, you can use this parameter as the default configuration item. In this way, you do not need to set this parameter for each job.

Configuring Periodic Scheduling

To configure the default action on the current job when the job it depends on fails, perform the following operations:

Step 1 In the navigation pane, choose **Configuration** > **Specifications**.

Step 2 Choose **Default Configuration**.

NOTE

Three options are available. The default value is **Terminate**.

- **Suspend**: The current job is suspended.
- **Continue**: The current job continues to be executed.
- **Cancel**: The current job is canceled.

Step 3 Click **Save** to save the settings. This parameter takes effect only for new jobs.

----End

Configuring the Multi-IF Policy

To configure the policy for executing nodes with multiple IF conditions, perform the following operations:

Step 1 In the navigation pane, choose **Configuration** > **Specifications**.

Step 2 Choose **Default Configuration**.

NOTE

The following two options are available:

- **OR**: Nodes are executed if an IF condition is met.
- **AND**: Nodes are executed if all IF conditions are met.

For details, see **Configuring the Policy for Executing a Node with Multiple IF Statements**.

Step 3 Click **Save** to save the settings.

----End

Configuring the Hard and Soft Lock Policy

The policy determines how you can grab the lock of a job or script. If you use a soft lock, you can grab the lock of a job or script regardless of whether you have the lock. If you use a hard lock, you can only unlock or grab the lock of a job or script for which you have the lock. Operations such as publish, execution, and scheduling are not restricted by locks.

You can configure the hard/soft policy based on your needs.

Step 1 In the navigation pane, choose **Configuration** > **Specifications**.

Step 2 Choose **Default Configuration**.

NOTE

The default policy is **Soft Lock**.

- **Soft lock**: You can lock or unlock jobs or scripts, regardless of whether they are locked by others.
- **Hard Lock**: You can lock jobs or scripts only after they have been unlocked by other users. The space administrator and the DAYU Administrator can lock and unlock jobs or scripts without any limitations.

Step 3 Click **Save** to save the settings.

----End

Configuring Script Variables

Variables of an SQL script can be in ${} or ${dlf.} format. You can configure either type as needed. The configured variable format applies to SQL scripts, SQL statements in jobs, single-node jobs, and environment variables.

Step 1 In the navigation pane, choose **Configuration** > **Configure**.

Step 2 Click **Default Configuration** and set **Script Variable Definition**.

NOTE

The default variable format is **${}**.

- **${}**: Identify the definition of the ${} format in the script and parse the field as the variable name. For example, variable name *xxx* is identified from **${***xxx***}**.
- **${dlf.}**: Identify the definition of the ${dlf.} format in the script and parse the **dlf.** field as the variable name. Other ${} format definitions are not recognized as variables. For example, variable name **dlf.***xxx* is identified from **${dlf.***xxx***}**.

Step 3 Click **Save** to save the settings.

----End

Configuring a Data Export Policy

By default, all users can download and dump the execution results of SQL scripts. If you do not want all users to have this permission, perform the following steps to configure a data export policy:

Step 1 In the navigation pane, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration** and set **Data Export Policy**.

NOTE

The default data export policy is **All User Can**.

- **All User Can**: All users can download and dump SQL execution results.
- **All User Cannot**: No user can download or dump SQL execution results.
- **Only Workspace Manager Can**: Only workspace administrators can download and dump SQL execution results.

Step 3 Click **Save**.

----End

Disabling Auto Node Name Change

On the **Develop Job** page, when you select a script for a node or associate a node with the function of another cloud service, the node name will be automatically changed to the script name or function name. You can disable this function.

Step 1 In the navigation pane, choose **Configuration** > **Specifications**.

Step 2 Choose **Default Configuration**. Find **Disable Auto Node Name Change** and select job nodes.

NOTE

- You can disable automatic name change for the following nodes: CDM Job, DIS Stream, DLI SQL, DWS SQL, MRS Spark SQL, MRS Hive SQL, MRS Presto SQL, MRS HetuEngine, MRS ClickHouse, MRS Impala SQL, Shell, RDS SQL, Subjob, For Each, or Python.
- No job nodes are selected by default.
- Names of the selected nodes will not be automatically changed when a script is selected or a function is associated with them.

Step 3 Click **Save**.

----End

Use Simple Variable Set

The simple variable set provides a series of customized variables to dynamically replace parameters during task scheduling.

Step 1 In the navigation pane on the **Data Development** page, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration** and set **Use Simple Variable Set**.

NOTE

- **Yes**: Simple variable sets are supported. A series of customized variables provided by the simple variable set. Customized parameters are automatically replaced with specific values based on the service date, plan time, and parameter value format of task scheduling. In this way, parameters can be dynamically replaced during task scheduling.
- **No**: Simple variable sets are not supported.

Step 3 Click **Save** to save the settings.

----End

Setting the Notification Policy for Jobs in Failure Ignored Status

To configure the notification type for jobs whose status is failure ignored, perform the following steps:

Step 1 In the navigation pane on the **Data Development** page, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration** and set **Notification Policy for Jobs in Failure Ignored Status**.

Step 3 Select a notification type for jobs whose status is failure ignored.

NOTE

- Jobs whose status is failure ignored are those whose **Policy for Handling Subsequent Nodes If the Current Node Fails** is set to **Go to the next node**. By default, such jobs are deemed successful by the system.
- You can configure either of the following notification types for such jobs:

  Abnormal

  Successful (default)

Step 4 Click **Save**.

----End

Setting Retry Node upon Timeout

You can set this parameter to specify whether a node will be re-executed if it fails upon timeout.

Step 1 In the navigation pane on the **Data Development** page, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration**.

Step 3 Set **Retry Node upon Timeout**.

NOTE

- **No**: A node will not be re-executed if it fails upon timeout.
- **Yes**: A node will be re-executed if it fails upon timeout.

Step 4 Click **Save** to save the settings.

----End

Exclude Waiting Time from Instance Timeout Duration

You can specify whether to exclude waiting time from instance timeout duration.

Step 1 In the navigation pane, choose **Configuration** > **Specifications**.

Step 2 Choose **Default Configuration** and set **Exclude Waiting Time from Instance Timeout Duration**.

Step 3 Select **Yes** or **No**.

NOTE

Yes: The waiting time before an instance starts running is excluded from the instance timeout duration.

No: The waiting time before an instance starts running is included in the instance timeout duration.

Step 4 Click **Save** to save the settings.

----End

Rules for Splitting MRS JAR Package Parameters

You can set the rule for splitting the string parameters (enclosed by "") in the JAR package parameters of MRS MapReduce and MRS Spark operators.

Step 1 In the navigation pane, choose **Configuration** > **Specifications**.

Step 2 Choose **Default Configuration** and set **Rules for Splitting MRS JAR Package Parameters**.

Step 3 Select a rule.

NOTE

Split String Arguments by Space: For example, **"select * from table"** is split into four parameters by space: **select**, *****, **from**, and **table**.

Do not split string arguments: For example, **"select * from table"** is regarded as one parameter and is not split.

Step 4 Click **Save** to save the settings.

----End

Synchronization of Job Version by Waiting Instance

You can specify whether a waiting instance can synchronize the latest job version.

Step 1 In the navigation pane, choose **Configuration** > **Specifications**.

Step 2 Choose **Default Configuration** and set **Synchronization of Job Version by Waiting Instance**.

Step 3 Select **Yes** or **No**.

NOTE

Yes: The waiting instance uses the latest job version.

No: The waiting instance still uses the existing job version.

Step 4 Click **Save** to save the settings.

----End

Execution Mode for Hive SQL and Spark SQL Statements

When Hive SQL and Spark SQL statements are executed, DGCDataArts Studio can place SQL statements in OBS or in the request body.

Step 1 In the navigation pane, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration**.

Step 3 Set **Execution Mode for Hive SQL and Spark SQL Statements**.

NOTE

In OBS: Hive SQL and Spark SQL statements are put in OBS, and the OBS is returned to MRS.

In the request message body: Hive SQL and Spark SQL statements are put in the request message body, and the script content is returned to MRS.

Step 4 Click **Save** to save the settings.

NOTE

This configuration supports Hive SQL and Spark SQL scripts, and pipeline and single-task jobs.

----End

Setting PatchData Priority

You can set the priority of a PatchData job. When system resources are insufficient, computing resources are preferentially allocated to jobs with higher priorities. A larger number indicates a higher priority. Currently, only the priorities of DLI SQL operators can be set.

Step 1 In the left navigation pane on the DataArts Factory console, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration** and set **PatchData Job Priority**.

Step 3 Set the patch data priority policy.

Step 4 Click **Save** to save the settings.

NOTE

The mapping between the **PatchData Job Priority** and **spark.sql.dli.job.priority** of DLI is as follows:

If **PatchData Job Priority** is set to **1**, **spark.sql.dli.job.priority** of DLI is **1**.

If **PatchData Job Priority** is set to **2**, **spark.sql.dli.job.priority** of DLI is **3**.

If **PatchData Job Priority** is set to **3**, **spark.sql.dli.job.priority** of DLI is **5**.

If **PatchData Job Priority** is set to **4**, **spark.sql.dli.job.priority** of DLI is **8**.

If **PatchData Job Priority** is set to **5**, **spark.sql.dli.job.priority** of DLI is **10**.

----End

Historical Job Instance Cancellation Policy

You can set the number of retention days for waiting job instances. If the waiting time of a job instance exceeds the configured retention days, the job instance is canceled. The minimum number of retention days is 2, that is, a job instance which is not executed can be canceled after at least two days. The default number of retention days is 60.

Step 1 In the left navigation pane on the DataArts Factory console, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration**.

Step 3 Set the number of retention days for waiting job instances.

Step 4 Click **Save** to save the settings.

----End

Send Alarm Upon Instance Cancellation If you select **Yes** for this parameter and configure a cancellation notification for a job, an alarm notification will be sent

when a historical job instance is canceled upon timeout. If you select **No**, no alarm notification will be sent.

Step 1 In the left navigation pane on the DataArts Factory console, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration**.

Step 3 Set **Send Alarm Upon Instance Cancellation**.

Step 4 Click **Save** to save the settings.

----End

Historical Job Instance Alarm Policy

You can set the number of days during which alarms can be generated for monitored job instances. The default value is seven days. Alarms cannot be sent for job instances beyond the seven-day period.

For example, if you set the value of this parameter to **2**, alarms can be generated for the job instances of yesterday and today, but cannot be generated for the job instances of the day before yesterday and of an earlier time even if the triggering conditions are met.

Step 1 In the navigation pane, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration** and locate **Historical Job Instance Alarm Policy**.

Step 3 Set the number of days during which alarms can be generated for monitored job instances.

NOTE

The default value is **7**. Set a value from 1 to 270.

After you set this parameter, alarms are generated only for the job instances which are created after this parameter is set and not for historical instances.

Step 4 Click **Save** to save the settings.

----End

Job Alarm Notification Topic

You can set the topic used to send notifications by owner.

Step 1 In the navigation pane, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration**.

Step 3 Set **Job Alarm Notification Topic**. Click **View Topic** to go to the SMN console to view available topics.

NOTE

You can only select a topic that you created on the SMN console (to prevent conflict with any existing topic). Only the workspace administrator can configure topic.

Step 4 Click **Save** to save the settings.

----End

Default Retry Policy upon Job Operator Failure

This policy takes effect only for new job operators in the current workspace. The default policy for the operators in historical jobs is not affected. The default value is **No**.

Step 1 In the left navigation pane on the DataArts Factory console, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration**.

Step 3 Set **Default Retry Policy upon Job Operator Failure**.

NOTE

If this parameter is set to **Yes**, new job operators can be retried once, and the retry interval is 120 seconds by default.

Step 4 Click **Save** to save the settings.

----End

Generate Alarm Upon Job Retry Failure

If you enable this function, an alarm is generated each time a job fails to be retried.

Step 1 In the left navigation pane on the DataArts Factory console, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration**.

Step 3 Set **Generate Alarm Upon Job Retry Failure**.

NOTE

- If you select **All jobs**, **Real-time jobs**, or **Batch jobs**, an alarm is generated each time a job fails to be retried.
- If you select **Disable**, an alarm is generated only when the maximum number of retries has been reached for the job.

Step 4 Click **Save** to save the settings.

----End

Automatic Script Name Transfer During Job Execution

If this function is enabled, **set mapreduce.job.name="*Script name*"** of the Hive SQL script is automatically transferred to MRS during job execution in the current workspace.

NOTE

This function takes effect only if the preceding parameter value has not been set for the script. If the parameter value has been set for the script, the value set is preferentially read and transferred to MRS. This function is unavailable for MRS clusters in security mode. To enable this function for such clusters, set them to non-security mode.

Step 1 In the navigation pane, choose **Configuration** > **Specifications**.

Step 2 Choose **Default Configuration**.

Step 3 Set **Automatic Script Name Transfer During Job Execution**.

NOTE

- **Yes**: The system automatically transfers the Hive SQL script name to MRS during job execution.
- No: The system does not automatically transfer the Hive SQL script name to MRS during job execution.

Step 4 Click **Save** to save the settings.

----End

Job Dependency Rule

Jobs can be depended on by jobs in other workspaces (requires the permission to query the job list in the workspace). All default roles in the workspace have this permission. Custom roles must have the job query permission in DataArts Factory.

Step 1 In the left navigation pane on the DataArts Factory console, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration**.

Step 3 Configure **Job Dependency Rule**.

NOTE

- **Jobs cannot be depended on by jobs in other workspaces**: Jobs in this workspace cannot be depended on by jobs in other workspaces.
- **Jobs can be depended on by jobs in other workspaces**: Jobs in this workspace can be depended on by jobs in other workspaces, without requiring the permissions of this workspace.
- **Jobs can be depended on by jobs in other workspaces (requires the permission to query the job list in the workspace)**: Jobs in this workspace can be depended on by jobs in other workspaces, requiring the permissions of this workspace. If you do not have the permissions, the system displays a message indicating that you do not have the permission to obtain the job list in workspace *xxx* when you configure job dependencies across workspaces.

Step 4 Click **Save** to save the settings.

----End

Script Execution History

You can set this parameter to control the permissions to view the script execution history.

Step 1 In the left navigation pane on the DataArts Factory console, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration**.

Step 3 Set **Script Execution History**.

NOTE

- **Myself**: The script execution history for only myself is displayed.
- **All users**: The script execution history for all users is displayed.

Step 4 Click **Save** to save the settings.

----End

Identity for Job Tests

After configuring this parameter, you can specify the identity used to test jobs.

Step 1 In the left navigation pane on the DataArts Factory console, choose **Configuration** > **Configure**.

Step 2 Choose **Default Configuration**.

Step 3 Set **Identity for Job Tests**.

NOTE

- **Public agency or IAM account**: A public agency or IAM account is used to execute jobs.
- **Personal account**: The user who clicks **Test** is used to execute jobs.

  If no workspace agency or IAM account is available, a personal account is used for job tests.

  If you are using a federated account, you must set this parameter to **Public agency or IAM account**.

Step 4 Click **Save** to save the settings.

----End

SparkSqlJob/Script Default Template Configuration

You can set this parameter to determine whether any parameters can be set to overwrite the default parameters of the template.

In the MRS API connection mode, default parameters can be configured for Spark SQL scripts. For proxy connections, this function is not supported.

Step 1 In the navigation pane, choose **Configuration** > **Specifications**.

Step 2 Choose **Default Configuration**.

Step 3 Set **SparkSqlJob/Script Default Template Configuration**.

NOTE

- **Yes**: You can set any parameters for jobs and scripts.
- **No**: You must select a template for jobs and scripts, and cannot overwrite the parameters in the template when configuring jobs and scripts. If you select **No**, select a default parameter template that has been configured. For details about how to configure a template, see **Configuring a Template**.

  Then go to the **basic information page of the Spark SQL job or Spark SQL script page and click** **in the upper right corner** to view the configured default program parameters. The preset default parameters are unavailable and cannot be modified.

  You can also customize program parameters. When a Spark SQL job or script is executed, the unavailable parameters in the template prevail.

----End

HiveSqlJob/Script Default Template Configuration

You can set this parameter to determine whether parameters can be set to overwrite the default parameters of the template.

In the MRS API connection mode, default parameters can be configured for Hive SQL scripts. For proxy connections, this function is not supported.

Step 1 In the navigation pane, choose **Configuration** > **Specifications**.

Step 2 Choose **Default Configuration**.

Step 3 Set **HiveSqlJob/Script Default Template Configuration**.

NOTE

- **Yes**: You can set any parameters for jobs and scripts.
- **No**: You must select a template for jobs and scripts, and cannot overwrite the parameters in the template when configuring jobs and scripts. If you select **No**, select a default parameter template that has been configured. For details about how to configure a template, see **Configuring a Template**.

  Then go to the **basic information page of the Hive SQL job or Hive SQL script page and click** **in the upper right corner** to view the configured default program parameters. The preset default parameters are unavailable and cannot be modified.

  You can also customize program parameters. When a Hive SQL job or script is executed, the unavailable parameters in the template prevail.

Step 4 Click **Save** to save the settings.

----End

Job/Script Change Management

If you enable this function, you can export job/script changes (addition, modification, and deletion) in a workspace to a .zip file, and import the file to another workspace.

Step 1 In the left navigation pane on the DataArts Factory console, choose **Configuration** > **Configure**.

Step 2 Click **Default Configuration**.

Step 3 Set **Job/Script Change Management**.

NOTE

- **Yes**: Events are recorded for job and script changes. All the changed jobs and scripts can be incrementally exported and imported by time.
- **No**: No events are recorded for job and script changes. Only selected jobs and scripts can be exported and imported.

Step 4 Click **Save** to save the settings.

NOTE

You can export and import jobs and scripts in the workspace only if you have set **Job/Script Change Management** to **Yes**.

----End

6.8.1.9 Configuring Task Groups

By configuring a task group, you can control the maximum number of concurrent nodes in a task group in a more fine-grained manner.

Constraints

This function is available only for batch processing jobs.

Task groups cannot be used across workspaces.

Procedure

Step 1 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 2 In the left navigation pane, choose **Configuration** > **Configure**.

Step 3 Choose **Task Groups**.

Step 4 Click **Create**.

Step 5 In the displayed dialog box, set required parameters.

Table 6-73 Parameters for creating a task group

Parameter	Description
Task Group Name	Name of the task group
Owner	Owner of the task group
Maximum number of concurrency	Maximum number of concurrent job nodes in the current task group The maximum number of concurrent nodes is the current number of concurrent DataArts Studio instances.
Description	Description of the task group

Step 6 Click **OK**.

After the task group is created, go to the job development page. On the canvas of a job, click **Scheduling Setup** and select the created task group. Then the number of concurrent nodes in the current task group can be controlled in a more fine-grained manner based on the selected task group.

----End

Follow-up Operations

Modifying a task group: Locate a task group and click **Modify** in the **Operation** column.

Deleting a task group: Locate a task group and click **Delete** in the **Operation** column. A task group used by a job cannot be deleted.

Viewing references: Locate a task group and click **View Reference** in the **Operation** column to view the jobs that are using the task group.

6.8.2 Managing Resources

You can upload custom code or text files as resources on Manage Resource and schedule them when running nodes. Nodes that can invoke resources include DLI Spark, MRS Spark, DLI Flink Job, and MRS MapReduce.

After creating a resource, configure the file associated with the resource. Resources can be directly referenced in jobs. When the resource file is changed, you only need to change the resource reference location. You do not need to modify the job configuration. For details about resource usage examples, see **Developing a DLI Spark Job**.

Constraints

This function depends on OBS or MRS HDFS.

(Optional) Creating a Directory

If a directory exists, you do not need to create one.

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the left navigation pane, choose **Configuration** > **Manage Resource**.
3. In the directory list, click  . In the displayed dialog box, configure directory parameters. **Table 6-74** describes the directory parameters.

Table 6-74 Resource directory parameters

Parameter	Description
Directory Name	Name of the resource directory. The name must contain 1 to 32 characters, including only letters, numbers, underscores (_), and hyphens (-).

Parameter	Description
Select Directory	Parent directory of the resource directory. The parent directory is the root directory by default.

4. Click **OK**.

Creating a Resource

You have enabled OBS before creating a resource.

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the left navigation pane, choose **Configuration** > **Manage Resource**.
3. Click **Create Resource**. In the displayed dialog box, configure resource parameters. **Table 6-75** describes the resource parameters. Click **OK**.

Table 6-75 Resource management parameters

Parameter	Mandatory	Description
Name	Yes	Name of the resource. The name must contain 1 to 32, including only letters, numbers, underscores (_), and hyphens (-).
Type	Yes	File type of the resource. Possible values: • jar: JAR file • pyFile: User Python file • file: User file • archive: User AI model file The supported file name extensions are **zip**, **tgz**, **tar.gz**, **tar**, and **jar**.
Resource Location	Yes	Location of the resource. OBS and HDFS are supported. HDFS supports only MRS Spark, MRS Flink Job and MRS MapReduce nodes.
File Path	Yes	Select an OBS file path when **Resource Location** is set to **OBS**. Select an MRS cluster name when **Resource Location** is set to **HDFS**.
Depended Package	No	This parameter is available only for DLI Spark nodes. Depended JAR package that has been uploaded to OBS. This parameter is required when **Type** is set to **jar** or **pyFile**.
Select Directory	Yes	Directory to which the resource belongs. The root directory is selected by default.

Parameter	Mandatory	Description
Description	No	Descriptive information about the resource.

Editing a Resource

After a resource is created, you can modify resource parameters.

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the left navigation pane, choose **Configuration** > **Manage Resource**.
3. In the **Operation** column of the resource, click **Edit**. In the displayed dialog box, modify the resource parameters. For details, see **Table 6-75**.
4. Click **OK**.

Deleting a Resource

You can delete resources that are no longer needed.

Before deleting a resource, ensure that it is not used by any jobs.

NOTICE

If you are trying to delete a resource that is being used by jobs, the **Delete Resource** dialog box is displayed. When you click **OK**, the **Reference List** dialog box is displayed, in which you can view the jobs that are using the resource and click **View** in the **Operation** column to go to the job details page.

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the left navigation pane, choose **Configuration** > **Manage Resource**.
3. In the **Operation** column of the resource, click **Delete**. The **Delete Resource** dialog box is displayed.
4. Click **Yes**.

Importing a Resource

To import a resource, perform the following operations:

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the left navigation pane, choose **Configuration** > **Manage Resource**.
3. In the resource directory, click and select **Import Resource**. The **Import Resource** dialog box is displayed.
4. Select the resource file that has been uploaded to OBS and click **Next**. After the import is complete, click **Close**.

Exporting a Resource

To export a resource, perform the following operations:

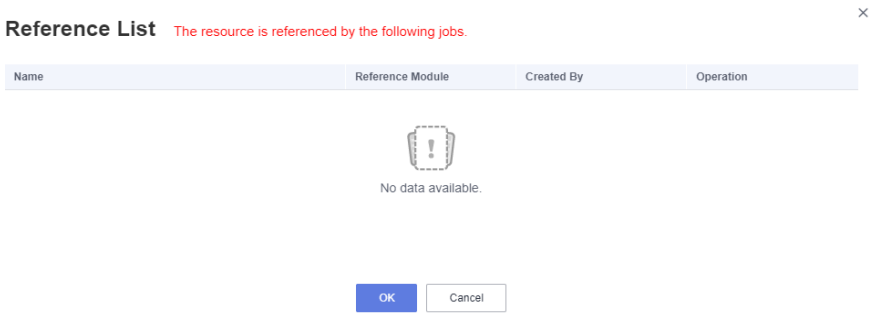
1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the left navigation pane, choose **Configuration** > **Manage Resource**.
3. In the resource directory, select a resource, click , and select **Export Resource**. The system starts downloading the resource to the local PC.

Viewing Resource References

To view the references of a resource, perform the following operations:

1. On the DataArts Studio console, locate a workspace and click **DataArts Factory**.
2. In the left navigation pane, choose **Configuration** > **Manage Resource**.
3. Right-click a resource in the list and select **View Reference**.
4. In the displayed **Reference List** dialog box, view the references of the resource.

Figure 6-118 Reference List dialog box



6.9 Review Center

For a workspace in simple mode, you can set the reviewer for the scripts and jobs you submit.

Constraints

- Only the admin of the current workspace can manage reviewers, including creating and deleting reviewers. The reviewer must be the admin of the current workspace or a user with the DAYU Administrator or Tenant Administrator permission.
- If the current workspace uses the enterprise mode, no application can be submitted for approval.
- You can only set whether to enable the review function and review jobs and scripts on the console.

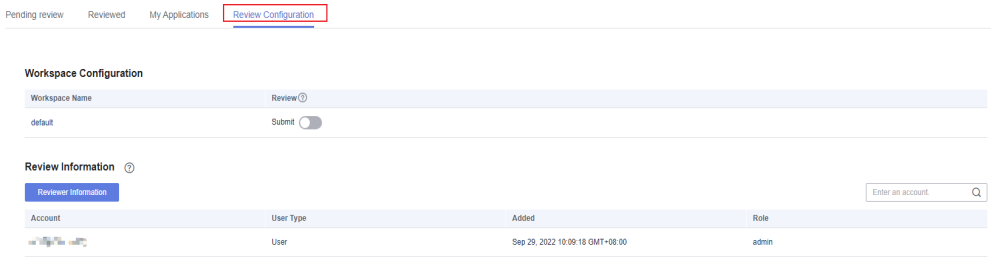
Approval Management

In the Review Center, you can view the applications you have submitted and their approval progress. If you are a reviewer, you can view the applications to be reviewed and the review history, manage reviewers, and enable or disable the review function.

- Pending Review
  a. In the left navigation pane, choose **Review Center**. In the right pane, click the **Pending Review** tab.

     On this page, you can view the applications that need to be reviewed.
  b. Click **Review** in the **Operation** column to view the application details and review the application.
  c. After entering the approval comments, approve or reject the application based on the actual situation.
- Reviewed
  a. In the left navigation pane, choose **Review Center**. In the right pane, click the **Reviewed** tab.
  b. Click **View Details** in the **Operation** column to view the review records and application content as an admin.
- My Applications
  a. In the left navigation pane, choose **Review Center**. In the right pane, click the **My Applications** tab.
  b. Click **View Details** in the **Operation** column to view details about an application.
  c. Click **Withdraw** in the **Operation** column to withdraw an application. You can submit the application again after modifying it.
- Review Settings

  Figure 6-119 Review Settings tab

  a. In the left navigation pane, choose **Review Center**. In the right pane, click the **Review Configuration** tab.
  b. On the **Review Configuration** page, you can enable or disable the review function as an admin. If the current workspace has applications that have not been reviewed, the review function cannot be disabled.

NOTE

- **Submit**: If you enable this function, you must specify an approver when submitting a job or script. To enable review for specified jobs/scripts, you must enable this function.
- **Specify jobs/scripts requiring review**: If you enable this function, you need to specify an approver only when submitting specified jobs or scripts. If you do not enable this function, all the jobs and scripts will require review. You need to configure the jobs or scripts that require approval. The procedure is as follows:

  Click the **Jobs Requiring Review** tab and then **Add from Baseline**. On the displayed page, select the priority jobs of the baseline task as the jobs that require review. Then click **OK**. The upstream jobs of the baseline also need to be approved.

  Click the **Scripts Requiring Review** tab and select the jobs corresponding to the baseline. The scripts associated with the jobs will be displayed on this page.

  If **Specify jobs/scripts requiring review** is disabled, all jobs and scripts need to be reviewed by default.
- If there are real-time pipeline jobs, **Submit** cannot be enabled.

c. Under **Reviewers**, you can view information about the reviewers in the current workspace as an admin.
   i. Click **Manage Reviewer**.
   ii. Locate the current workspace and click **Edit** in the **Operation** column.
   iii. Next to **Workspace Members**, click **Add**.
   iv. Search for and select a member account and select the **admin** role for it.
   v. Click **OK**.

6.10 Download Center

You can download or dump SQL script execution results. After downloading or dumping the SQL execution results, you can view them on the **Download Center** page.

Constraints

Records are generated on the **Download Center** page only when SQL scripts or single-task SQL jobs are executed, and results are downloaded or dumped.

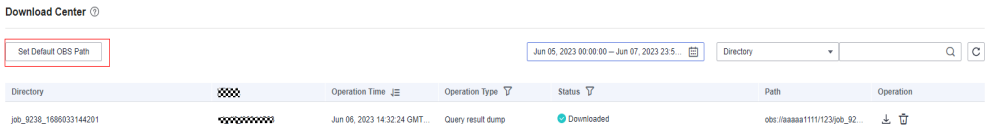
Download Center

NOTE

- The download records age out on a regular basis. When aged out, download records and the data dumped to OBS are deleted.
- Operators can view only their own download records. Workspace admins can view all download records in the current workspace.

On the **Download Center** page, you can centrally manage the execution results of SQL scripts. You can view and delete the download results, and view, download, and delete the dump results.

Figure 6-120 Download Center



- Set the default OBS path.

  NOTE

  The workspace admin can set the default OBS path for dump for the current workspace.

  a. In the left navigation pane on the DataArts Factory console, choose **Download Center**.
  b. Click **Set Default OBS Path**.
  c. Set the default OBS path.

     NOTE

     After you set the default OBS path, the test running results of scripts or single-task jobs will be dumped to this path by default. However, the paths where previous running results have been dumped will not change.

  d. Click **OK**.

- View the script execution result.

  a. In the left navigation pane on the DataArts Factory console, choose **Download Center**.
  b. View the file name, operator, operation time, operation type, task status, and OBS path of local download tasks and asynchronous dump tasks.

     You can view the dump task download failure records.

  c. Click in the **Operation** column to download data from the OBS path.
  d. Click in the **Operation** column to delete download and dump records.

     When you click **Delete**, a message is displayed indicating that the record cannot be downloaded after being deleted. Click **OK**.

- Filter records by search criteria.

  You can filter records by operation time, job name, OBS path, operator, operation type, and task status. You can enter a keyword for fuzzy search.

6.11 Node Reference

6.11.1 Node Overview

A node defines the operations performed on data. DataArts Factory provides nodes used for data integration, computing and analysis, database operations, and resource management. You can choose your desired nodes

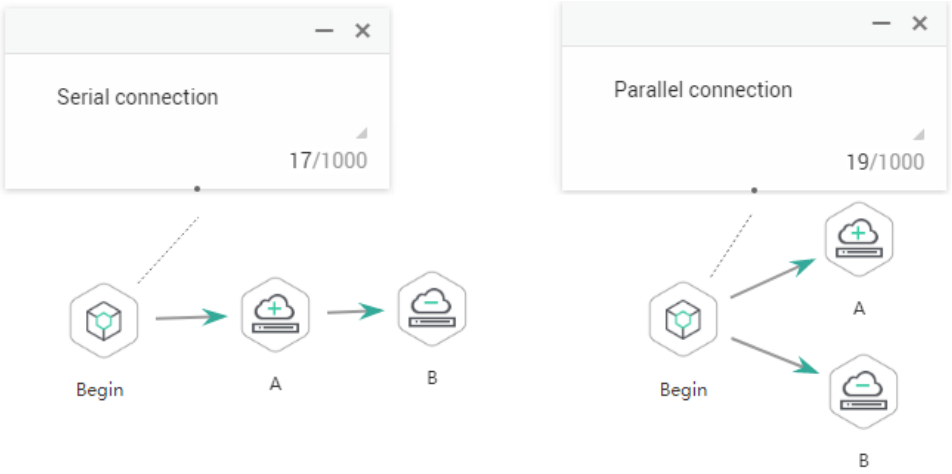
- Node parameters can be presented using Expression Language (EL). For details about how to use EL, see **Expression Overview**.

- Nodes cannot be connected in serial or parallel mode.

  Serial connection: Nodes are run one by one. Specifically, node B runs only after node A is finished running.

  Parallel connection: Nodes are run at the same time.

  Figure 6-121 Connection diagram



6.11.2 Node Lineages

6.11.2.1 Overview

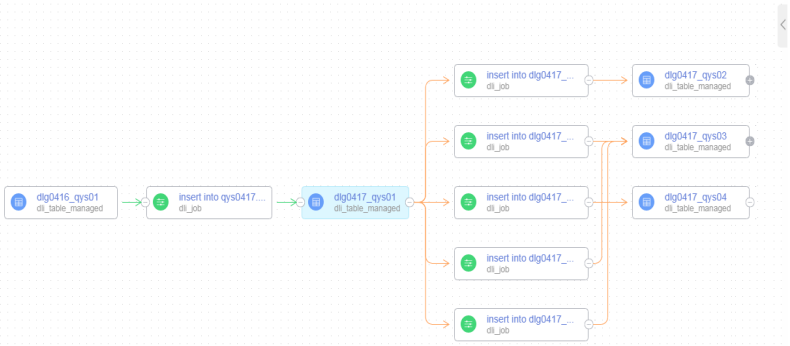
What Is Data Lineage?

In the era of big data, various types of data are rapidly generated due to explosive data growth. The massive and complex data information is converged, transformed, and transferred to generate new data and aggregate into an ocean of data.

During this process, a relationship is formed between the data, and these relationships are their lineages. They are analogous to the genetic relationships between people. However, in contrast from our human lineages, data lineages have the following distinct features:

- **Belongingness**: Specific data belongs to a specific organization or individual.
- **Multi-source**: One piece of data can have multiple sources. One piece of data may be generated by processing multiple pieces of data, and there may be multiple such processes.
- **Traceability**: The data lineage is traceable. It reflects the data lifecycle and the entire process from data generation to data disappearance.
- **Hierarchy**: The data lineage is hierarchical. Data classification and summary form new data, and different levels of description result in data layers.

Figure 6-122 Data lineage example



How DataArts Studio Data Lineage Is Implemented

- Generation of data lineages:

  The DataArts Studio data lineage parsing solution supports automatic lineage analysis and manual lineage configuration. Automatic lineage parsing is recommended. In this mode, lineages can be generated without manual configuration. If automatic lineage parsing is not supported, manually configure lineages.

  - Automatic lineage parsing: Lineages are automatically generated after the system parses the data processing and data migration nodes in data development jobs. No manual configuration is required. For details about the node types and scenarios that support automatic lineage parsing, see **Automatic Lineage Parsing**.
  - Manual lineage configuration: Customize the input and output tables of lineages in data development job nodes. If you configure lineages manually for a node, the automatic lineage parsing does not take effect for this node. For details about the node types that support manual lineage configuration, see **Manually Configuring a Lineage**.

- Display of data lineages:

  You need to create a metadata collection task in DataArts Catalog first. When a data development job meets the **automatic lineage parsing requirements** or **lineages have been manually configured**, and when the job is successfully scheduled, you can view the data lineages in DataArts Catalog.

6.11.2.2 Configuring Data Lineages

The DataArts Studio data lineage parsing solution supports automatic lineage analysis and manual lineage configuration. Automatic lineage parsing is recommended. In this mode, lineages can be generated without manual configuration. If automatic lineage parsing is not supported, manually configure lineages.

- Automatic lineage parsing: Lineages are automatically generated after the system parses the data processing and data migration nodes in data development jobs. No manual configuration is required. For details about the node types and scenarios that support automatic lineage parsing, see **Automatic Lineage Parsing**.
- Manual lineage configuration: Customize the input and output tables of lineages in data development job nodes. If you configure lineages manually

for a node, the automatic lineage parsing does not take effect for this node. For details about the node types that support manual lineage configuration, see **Manually Configuring a Lineage**.

Constraints

Currently, field-level lineage parsing is not supported.

Automatic Lineage Parsing

Automatic lineage parsing does not require manual configuration. When a data development job contains the nodes and scenarios listed in **Table 6-76**, the system can automatically parse lineages.

NOTE

The lineage of an SQL node can be parsed using multiple SQL statements, and column-level lineage parsing is supported. A single SQL statement cannot contain semicolons (;).

Table 6-76 Job nodes and scenarios that support automatic lineage parsing

Job Node	Supported Scenario
DLI SQL	• Lineages generated by data insertion between DLI tables • Lineages between OBS files generated by table creation statements and DLI tables
DWS SQL	Lineages between DWS tables generated by DML operations such as "Insert into"
MRS Hive SQL	Lineages between MRS tables generated by DML operations such as "Insert into/overwrite"
MRS Spark SQL	Lineages between MRS tables generated by DML operations such as "Insert into/overwrite"
CDM Job	Lineages generated during table file migration between MRS Hive, DLI, RDS, CSS, DWS, and OBS
ETL Job	Data lineages generated by ETL tasks between DLI, OBS, MySQL, and DWS.

Manually Configuring a Lineage

In a DataArts Studio data development job, you can customize the input and output tables of lineages on the nodes of the job. If you configure lineages manually for a node, the automatic lineage parsing does not take effect for this node.

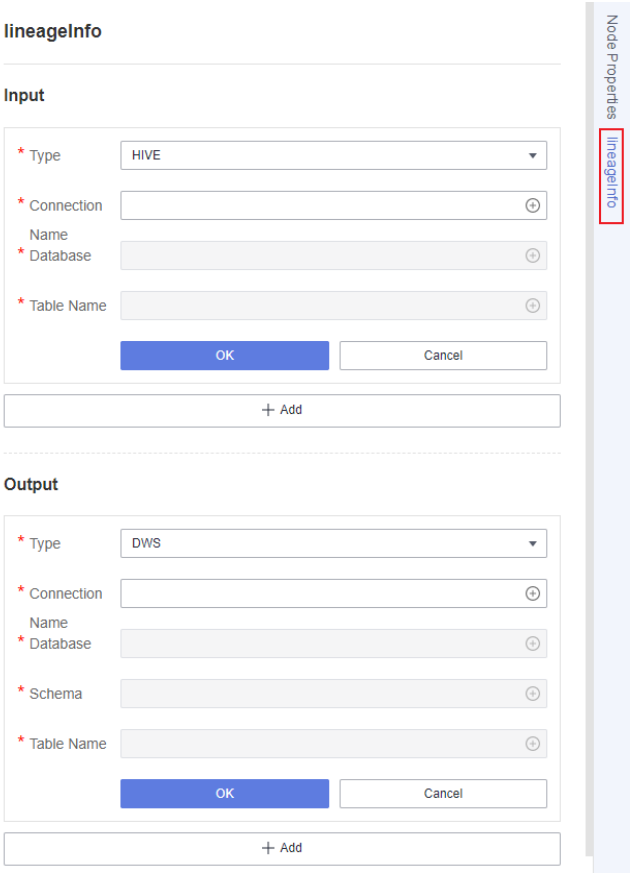
The following types of job nodes support manual lineage configuration.

- **CDM Job**

- **Rest Client**
- **DLI SQL**
- **DLI Spark**
- **DWS SQL**
- **MRS Spark SQL**
- **MRS Hive SQL**
- **MRS Presto SQL**
- **MRS Spark**
- **MRS Spark Python**
- **ETL Job**
- **OBS Manager**

When manually configuring the lineage, configure the input and output tables of the lineage on the Lineage tab page of the node. The data sources of the input and output tables can be DLI, DWS, Hive, CSS, OBS and CUSTOM. CUSTOM indicates a custom type. When manually configuring a lineage, you can add data sources that are not supported as custom types.

Figure 6-123 Example of manual configuration of lineage relationships

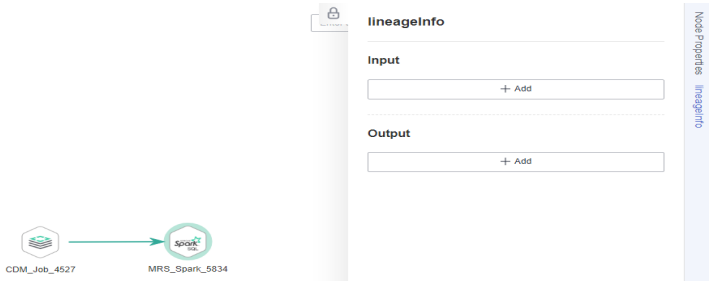


For example, you need to manually configure a lineage for an MRS Spark node in a pipeline data development job because this node does not support automatic lineage parsing. The procedure is as follows:

Step 1 On the DataArts Factory console, choose **Data Development** > **Develop Job**. Double-click the name of the job for which you want to configure a lineage to open the job canvas.

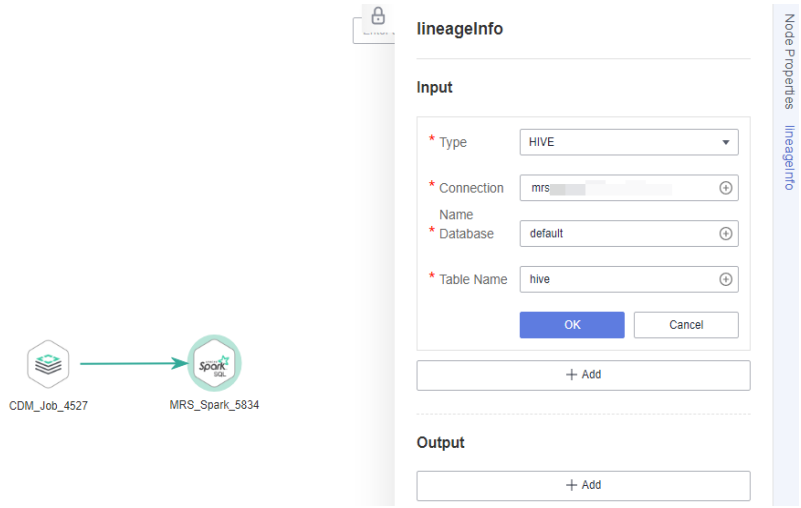
Step 2 Click the MRS Spark node in the job canvas and then the **lineageInfo** page.

Figure 6-124 lineageInfo page



Step 3 Configure the lineage input table. For example, you can configure input table **hive**, as shown in **Figure 6-125**.

Figure 6-125 Configuring the lineage input



Step 4 Click **OK** and configure the lineage output table. For example, you can configure output table **a**, as shown in **Figure 6-126**.

Figure 6-126 Configuring the lineage output

Step 5 Click **OK**. The lineage for the MRS Spark node has been configured. If you want to view the lineage later, collect metadata by referring to **Viewing Data Lineages** and schedule the job. Then, you can view the manually configured lineage of the MRS Spark node in DataArts Catalog.

----End

6.11.2.3 Viewing Data Lineages

You need to create a metadata collection task in DataArts Catalog first. When a data development job meets the **automatic lineage parsing requirements** or **lineages have been manually configured**, and when the job is successfully scheduled, you can view the data lineages in DataArts Catalog.

Constraints

- Data lineage updates depend on job scheduling. Data lineages are generated based on the latest job instances.
- To delete data lineages, you need to delete jobs or job metadata. Stopping jobs alone does not delete data lineages.

Creating and Running a Metadata Collection Task

Create and run a metadata collection task by referring to **Task Management**. When creating the task, select the tables whose lineages you want to view.

If a task for collecting the metadata of these tables has been created and run, skip this part.

Starting Job Scheduling

After metadata is collected, the system generates data lineages based on the latest job instances.

Step 1 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

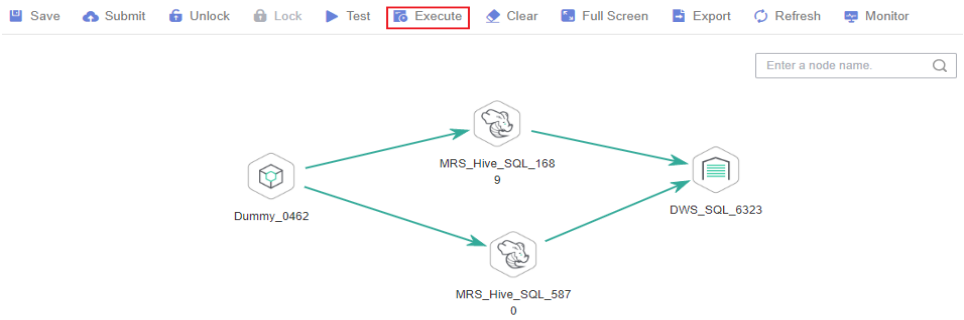
Step 2 In the navigation pane, click  and double-click the job for which lineages have been configured to open it.

Step 3 Click **Execute**. The system starts parsing lineages of the job.

NOTE

If you click **Test**, the system will not parse lineages of the job.

Figure 6-127 Starting job scheduling



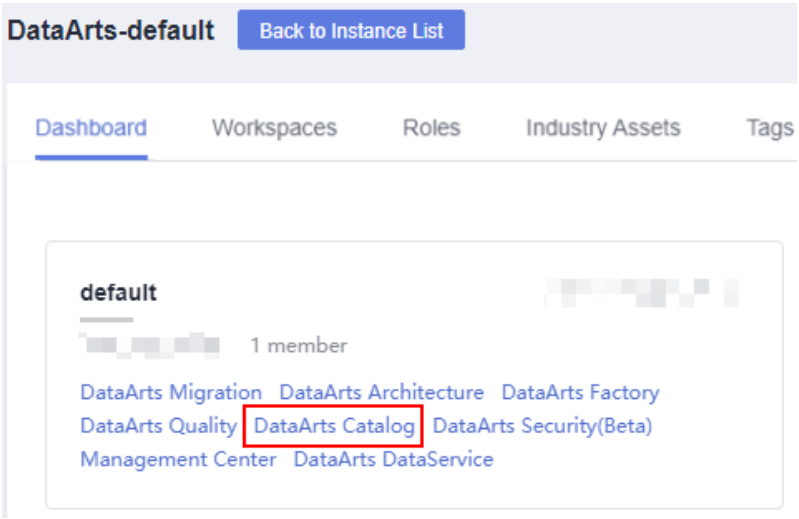
Step 4 After the job is successfully executed, wait for about 1 minute. The data lineage is generated.

----End

Viewing Data Lineages

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

Figure 6-128 DataArts Catalog



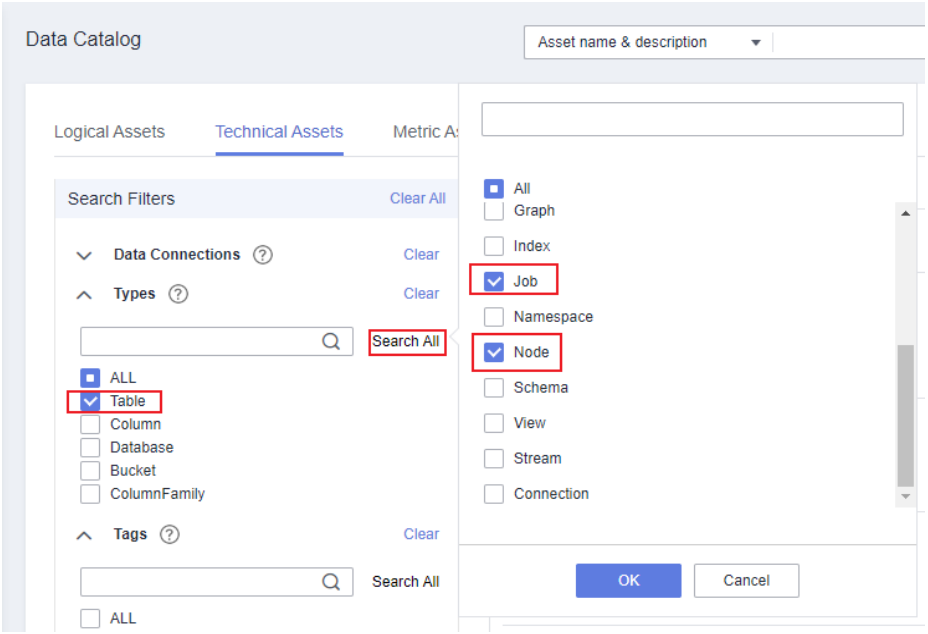
Step 2 In the navigation pane, choose **Data Catalog**. In the right pane, click the **Technical Assets** tab. On this page, you can query jobs, nodes, and tables.

In the **Types** area, click **Search All**, select **Job**, **Node**, and **Table**, and click **OK**.

NOTE

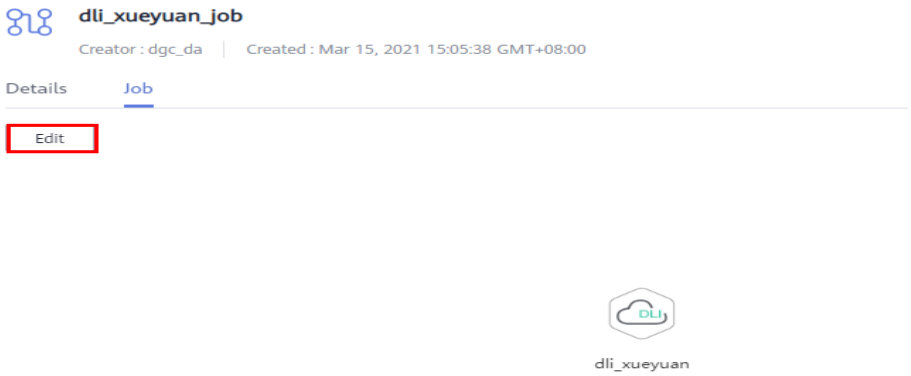
Jobs do not belong to any data connection. If you select a data connection in the search filters, no result will be returned.

Figure 6-129 Selecting types



Step 3 In the search result, click the name of an asset ending with **_job** to view its details. On the job details page, click the **Job** tab and then **Edit** to go to the job editing page.

Figure 6-130 Viewing job details

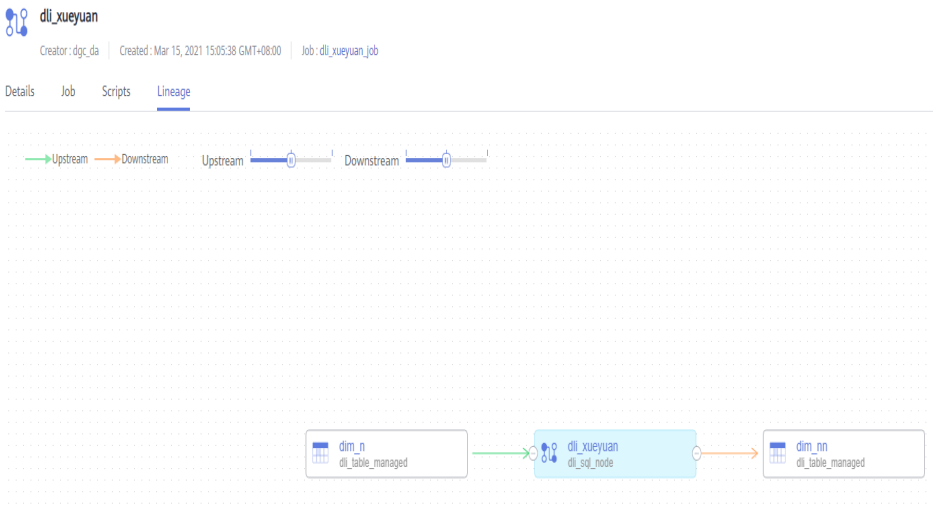


Step 4 In the data asset search result, click the name of an asset ending with **_node** to view its details. On the node details page, you can view the node lineage information.

- Click the **+** or **-** icon beside the node to expand its upstream and downstream links.

- Click a node to view the its details.
- Click the **Job** tab and then **Edit** to go to the job editing page.

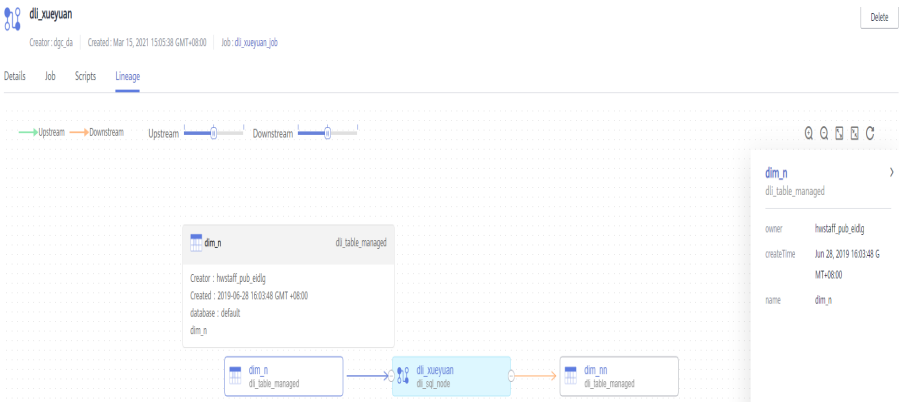
Figure 6-131 Viewing lineages of a node



Step 5 In the data asset search result, click the name of an asset whose icon is a table to view its details. On the table details page, you can view lineages of the table.

- Click the **+** or **-** icon beside the table to expand its upstream and downstream links.
- Click a table to view the its details.

Figure 6-132 Viewing lineages of a table



----End

6.11.3 CDM Job

Functions

The CDM Job node is used to run a predefined CDM job for data migration.

NOTE

If you have configured a macro variable of date and time in a CDM job and schedule the CDM job through DataArts Factory, the system replaces the macro variable of date and time with (*Planned start time of the data development job – Offset*) rather than (*Actual start time of the CDM job – Offset*).

Parameters

Table 6-77, **Table 6-78**, and **Table 6-79** describe the parameters of the CDM Job node. Configure the lineage to identify the data flow direction, which can be viewed in the DataArts Catalog module.

Table 6-77 Parameters of CDM Job nodes

Parameter	Mandatory	Description
CDM Cluster Name	Yes	Name of the CDM cluster to which the CDM job to be executed belongs. You can select two CDM clusters to improve job reliability. • If you select two clusters, they are delivered randomly to share load. If one cluster is abnormal, jobs are switched to the other cluster. • If you select two clusters, you are advised to set **Job Type** to **Existing jobs** rather than **New jobs** and ensure that the job exists in both clusters. You can create a CDM job in one cluster, export it, and import it to the other cluster to implement job synchronization. For details, see **Exporting and Importing CDM Jobs in Batches**.

Job Type	Yes	• Existing jobs
		• New jobs
		NOTE
		• If **Job Type** is **Existing jobs**, the job node is not updated when the CDM job is modified. To update the job node, save the job where the node is located again to trigger a CDM job update.
		• If **Job Type** is **New jobs**, the system checks whether a CDM job with the same name is running.
		  • If the CDM job is not running, update the job with the same name based on the request body.
		  • If a CDM job with the same name is running, update the job after the job is run. During this period, the job may be started by other tasks. As a result, the extracted data may not be the same as expected (for example, the job configuration is not updated, or the macro of the running time is not correctly replaced). Therefore, do not create multiple jobs with the same name.

Parameter	Mandatory	Description
CDM Job Name	No	This parameter is required only when **Job Type** is set to **Existing jobs**. Name of the CDM job to be executed. If the CDM job uses the **job parameters** or **environment variables** configured during data development, data can be indirectly migrated based on the parameters or variables during node scheduling in the DataArts Factory module.
CDM Job Message Body	No	This parameter is required only when **Job Type** is set to **New jobs**. Enter the JSON message body of the CDM job. For convenience, you can choose **More** > **View Job JSON** in the **Operation** column of an existing CDM job, copy the JSON content, and modify the content here. If the CDM job uses the **job parameters** or **environment variables** configured during data development, data can be indirectly migrated based on the parameters or variables during node scheduling in the DataArts Factory module.
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>). By default, the node name is the same as that of the selected CDM job. If you want the node name to be different from the CDM job name, disable this function by referring to **Disabling Auto Node Name Change**.

Table 6-78 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	Specifies how often the system check completeness of the node task. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	indicates the execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. • **Yes**: The node task will be re-executed, and the following parameters must be configured: – Maximum Retries – Retry Interval (seconds) • **No**: The node will not be re-executed. This is the default setting. **NOTE** • You are advised to configure automatic retry for only file migration jobs or database migration jobs with **Import to Staging Table** enabled to avoid data inconsistency caused by repeated data writes. • If parameter transfer is used for scheduling the CDM job, do not configure parameter **Retry upon Failure** in the CDM job. • If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Table 6-79 Lineage

Parameter	Description
Input	

Parameter	Description
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the output lineage.

6.11.4 DIS Stream

Functions

The DIS Stream node is used to query the status of a DIS stream. If the DIS stream is normal, you can perform other nodes. If the DIS stream is abnormal, the DIS Stream node will send an error message and exit. If you want to perform other nodes, you must set **Failure policy** to **Proceed to the next node**. For details about how to set **Failure policy**, see **Table 6-81**.

Parameters

Table 6-80 and **Table 6-81** describe the parameters of the DIS Stream node.

Table 6-80 Parameters of DIS Stream nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>). By default, the node name is the same as that of the selected stream. If you want the node name to be different from the stream name, disable this function by referring to **Disabling Auto Node Name Change**.
Stream Name	Yes	Select or enter the DIS stream to query. When entering the stream name, you can reference job parameters and use the EL expression. For details, see **Expression Overview**. To create a DIS stream, you can use either of the following methods: • Click to go to the **Data Integration** page and create a DIS stream on the **Stream Management** page. • Go to the DIS console to create a DIS stream.

Table 6-81 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.5 DIS Dump

Functions

The DIS Dump node is used to configure data dump tasks in DIS.

Parameters

Table 6-82 and **Table 6-83** describe the parameters of the DIS Dump node.

Table 6-82 Parameters of DIS Dump nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Stream Name	Yes	Select or enter the DIS stream to query. When entering the stream name, you can reference job parameters and use the EL expression. For details, see **Expression Overview**. To create a DIS stream, you can use either of the following methods: • Click . On the **Stream Management** page of DLF, create a DIS stream. • Go to the DIS console to create a DIS stream.
Duplicate Name Policy	Yes	Select a duplicate name policy. If the name of a dump task already exists, you can adopt either of the following policies based on site requirements: • Ignore: Give up adding the dump task and exit DIS Dump. The status of DIS Dump is **Succeeded**. • Overwrite: Continue to add the dump task by overwriting the one with the same name.

Parameter	Mandatory	Description
Dump Destination	Yes	1. Destination to which data is dumped. Possible values: • **OBS**: After the streaming data is stored to DIS, it is then periodically imported to OBS. After the real-time file data is stored to DIS, it is imported to OBS immediately. 2. Click  . In the dialog box that is displayed, set dump parameters. For details, see "Managing a Dump Task" in *Data Ingestion Service User Guide*.

Table 6-83 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.

Parameter	Mandatory	Description
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.6 DIS Client

Functions

The DIS Client node is used to send messages to a DIS stream.

You can learn more about how to use the DIS Client node in **Scheduling Jobs Across Workspaces**.

Parameters

Table 6-84 describes the parameters of the DIS Client node.

Table 6-84 Parameters of DIS Client nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Using DIS Connection	No	If DIS streams are used, messages can be sent to the DIS streams of another account. Otherwise, messages can be sent only to streams in all regions of the current account.
DIS Connection	No	This parameter is mandatory only when **Using DIS Connection** is set to **Yes**. Before setting this parameter, ensure that you have created a DIS connection in the Management Center by referring to **Creating Data Connections**. This parameter is not required when **Using DIS Connection** is set to **No**.
Region	No	Region that the target DIS stream belongs to. The DIS Client node is used to send messages to the target DIS stream.
Stream Name	Yes	DIS stream to which messages will be sent. You can enter a stream address or select a stream.
Sent Data	Yes	Text sent to the DIS stream. You can directly enter text or click  to use the EL expression.
Related Job	No	Select batch or real-time processing jobs. You can select a maximum of 10 jobs. This parameter allows you to switch to the monitoring page of the selected jobs when they start running. After selecting a job, click **Monitor**. On the **Monitory Job** page, select the DIS Client node and click **View Related Job** on the lower part of the page. In the **View Related Job** dialog box, click **View** in the **Operation** column of a job to view the details about the job.

Table 6-85 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.

Parameter	Mandatory	Description
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.7 Rest Client

Functions

The Rest Client node is used to respond to RESTful requests in Huawei Cloud.

For details about how to use the Rest Client operator, see **Obtaining the Return Value of a Rest Client Node**.

NOTE

If some APIs of the Rest Client node cannot be called due to network restrictions, you can use a shell script to call the APIs. To call an API using a shell script, you must have an ECS that can communicate with the API. Create a host connection and run the curl command to call the API using the shell script.

Rest Client operators do not support response bodies larger than 30 MB.

Parameters

Table 6-86, **Table 6-87**, and **Table 6-88** describe the parameters of the Rest Client node.

Table 6-86 Parameters of Rest Client nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).

Parameter	Mandatory	Description
Agent Name	Yes	Name of a CDM cluster. The CDM cluster provides the agent connection function. If the selected CDM cluster is in the same VPC as the third-party service, the REST client can call APIs on the tenant plane. **NOTE** You can select multiple clusters and must ensure that at least one cluster can be connected. If multiple clusters can be connected, DataArts Factory randomly connects one.
URL Address	Yes	IP address or domain name and port number of the request host. For example: https://192.160.10.10:8080
HTTP Method	Yes	Type of the request. Possible values: • **GET** • **POST** • **PUT** • **DELETE**
API Authentication Mode	Yes	• **IAM**: APIs can be accessed only by cloud users. The request header of a message sent by DataArts Studio to an API contains the authentication information of the current user. • **Non-authentication**: Authentication is not required for calling APIs. • **Username/Password**: The API caller needs to enter the username and password. When the DataArts Studio service sends a message, the request header contains the **Authorization** field. **NOTE** If the username and password authentication mode is used, you need to select a data connection that supports username and password authentication.
Request Header	No	Click + to add a request header. The parameters are described as follows: • Parameter Name Name of a parameter. The options are **Content-Type** and **Accept-Language**. • Parameter Value Value of the parameter

Parameter	Mandatory	Description
URL Parameter	No	Enter a URL parameter. The value is a character string in **key=value** format. Character strings are separated by newlines. This parameter is available only when **HTTP Method** is set to **GET**. Set these parameters as follows: ● Parameter The parameter contains a maximum of 32 characters, including only letters, numbers, hyphens (-), and underscores (_). ● Value The value contains a maximum of 64 characters, including only letters, digits, hyphens (-), underscores (_), number signs (#), open braces ({), and close braces (}).
Request Body	Yes	The request body is in JSON format. This parameter is available only when **HTTP Method** is set to **POST** or **PUT**.
Check Return Value	No	Checks whether the value of the returned message is the same as the expected value. This parameter is available only when **HTTP Method** is set to **GET**. Possible values: ● **YES**: Check whether the return value is the same as the expected one. ● **NO**: No need to check whether the return value is the same as the expected one. A 200 response code is returned (indicating that the node is successfully performed).

Parameter	Mandatory	Description

Property Path	Yes	Path of the property in the JSON response message. Each Rest Client node can have only one property path. This parameter is available only when **Check Returned Value** is set to **YES**.
		For example, the returned result is as follows:
		{
		"param1": "aaaa",
		"inner":
		{
		"inner":
		{
		"param4": 2014247437
		},
		"param3": "cccc"
		},
		"status": 200,
		"param2": "bbbb"
		}
		The **param4** path is **inner.inner.param4**.
		You can also learn how to configure this parameter by referring to **Obtaining the Return Value of a Rest Client Operator**.

Request Success Flag	Yes	Enter the request success flag. If the returned value of the response matches one of request success flags, the node is successfully performed. This parameter is available only when **Check Returned Value** is set to **YES**. The request success flag can contain only letters, numbers, hyphens (-), underscores (_), dollar signs ($), open braces ({), and close braces (}). Separate values with semicolons (;).
Request Failure Flag	No	Enter the request failure flag. If the returned value of the response matches one of request failure flags, the node is successfully performed. This parameter is available only when **Check Returned Value** is set to **YES**. The request failure flag can contain only letters, numbers, hyphens (-), underscores (_), dollar signs ($), open braces ({), and close braces (}). Separate values with semicolons (;).

Parameter	Mandatory	Description
Retry Interval (seconds)	Yes	If the return value of the response message does not match the request success flag, the node keeps querying the matching status at a specified interval until the return value of the response message is the same as the request success flag. By default, the timeout interval of the node is one hour. If the return value of the response message does not match the request success flag within this period, the node status changes to **Failed**. This parameter is available only when **Check Returned Value** is set to **YES**.
The response message body parses the transfer parameter.	No	Specify the mapping between the job variable and JSON property path. Separate parameters by newline characters. For example: var4=inner.inner.param4 **var4** is a job variable. The job variable must contain 1 to 64 characters, including only letters and numbers. **inner.inner.param4** is the JSON property path. This parameter takes effect only when it is referenced by the subsequent node. When this parameter is referenced, the format is **${var4}** **NOTE** The variable name (for example, **var4**) must be unique in the current job.

Table 6-87 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

Table 6-88 Lineage

Parameter	Description
Input	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the output lineage.

6.11.8 Import GES

Function

The Import GES node is used to import files from an OBS bucket to a GES graph.

Parameters

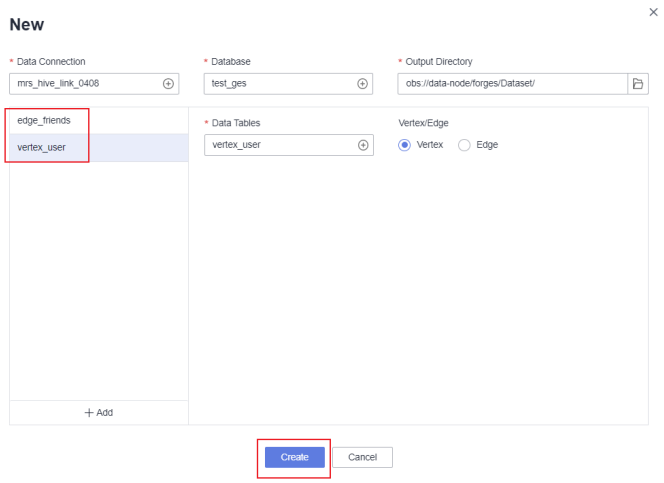
Table 6-89 and **Table 6-90** describe the parameters of the Import GES node.

Table 6-89 Parameters of Import GES nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Graph Name	Yes	You can directly select the graph to import or manually enter the graph name. To create a GES graph, go to the GES console.
Metadata Source	Yes	Two types of metadata sources are available: • **Existing file**: Select an existing XML metadata file from an OBS bucket. • **New**: Generate an XML metadata file in an OBS bucket based on the vertex tables and edge tables in MRS Hive. **NOTE** Set at least one of the following parameters: **Metadata**, **Edge Data Set**, and **Vertex Data Set**.

Parameter	Mandatory	Description

Metadata	No	Set this parameter based on the value you select for **Metadata Source**.
		• If you select **Existing file** for **Metadata Source**, click in the text box and select the corresponding metadata file.
		• If you select **New** for **Metadata Source**, click in the text box. In the displayed dialog box, select the vertex table and edge table in MRS Hive, enter the OBS path for storing the metadata, and click **Create**. Then the system automatically generates an XML metadata file and saves it to the OBS path you enter.
		The vertex table and edge table in MRS Hive are the edge data set and vertex data set normalized based on the GES graph data format. They must be consistent with the values of **Edge Data Set** and **Vertex Data Set**, respectively.
		The vertex and edge data sets must comply with the data format requirements of GES graphs. The graph data format requirements are briefed as follows. For details, see **Graph Data Formats**.
		– The vertex data set contains the data of each vertex. Each row is the data of a vertex. The format is as follows. **id** is the unique identifier of vertex data.
		`id,label,property 1,property 2,property 3,...`
		– The edge data set contains the data of each edge. Each row is the data of an edge. Graph specifications in GES are defined based on the edge quantity, for example, one million edges. The format is as follows. **id 1** and **id 2** are the IDs of the two endpoints of an edge.
		`id 1, id 2, label, property 1, property 2,...`

Parameter	Mandatory	Description
		NOTE When creating metadata, note the following: 1. You can only select a vertex table and a edge table that use a single label. If you select a vertex table or a edge table that has multiple labels, the generated metadata may be missing. 2. The metadata XML file is generated after you click **Create**. If the structure of the vertex table and edge table changes during subsequent job scheduling, the metadata XML file will not be updated automatically. In this case, you need to open the **New** dialog box and click **Create** again to generate a new metadata XML file. 3. In the generated metadata XML file, the value of **Cardinality** (data composite type) in **Property** is **single** and cannot be changed. 4. You can generate metadata XML files for multiple pairs of vertex tables and edge tables at a time. However, only one table can be selected for the **Edge Data Set** and **Vertex Data Set** parameters of the Import GES node. If there are multiple pairs of vertex tables and edge tables, you are advised to create metadata XML files on multiple Import GES nodes. In this way, you can ensure that each piece of metadata corresponds to each pair of vertex tables and edge tables during the import of graph data. **Figure 6-133** New 

Parameter	Mandatory	Description

Edge Data Set	No	You can select the edge data set CSV file in the corresponding OBS bucket or select the OBS path of the edge data set.
		The vertex and edge data sets must comply with the data format requirements of GES graphs. The graph data format requirements are briefed as follows. For details, see **Graph Data Formats**.
		• The vertex data set contains the data of each vertex. Each row is the data of a vertex. The format is as follows. **id** is the unique identifier of vertex data.
		`id,label,property 1,property 2,property 3,...`
		• The edge data set contains the data of each edge. Each row is the data of an edge. Graph specifications in GES are defined based on the edge quantity, for example, one million edges. The format is as follows. **id 1** and **id 2** are the IDs of the two endpoints of an edge.
		`id 1, id 2, label, property 1, property 2,...`

Vertex Data Set	No	You can directly select the corresponding Vertex data set or select the OBS path of the Vertex data set.
		The vertex and edge data sets must comply with the data format requirements of GES graphs. The graph data format requirements are briefed as follows. For details, see **Graph Data Formats**.
		• The vertex data set contains the data of each vertex. Each row is the data of a vertex. The format is as follows. **id** is the unique identifier of vertex data.
		`id,label,property 1,property 2,property 3,...`
		• The edge data set contains the data of each edge. Each row is the data of an edge. Graph specifications in GES are defined based on the edge quantity, for example, one million edges. The format is as follows. **id 1** and **id 2** are the IDs of the two endpoints of an edge.
		`id 1, id 2, label, property 1, property 2,...`

Edge Processing	Yes	The edge processing supports the following modes: • Allow repetitive edges • Ignore subsequent repetitive edges • Overwrite previous repetitive edges

Parameter	Mandatory	Description
Offline	No	Whether offline import is used. The value is **Yes** or **No**, and the default value is **No**. • **true**: Offline import is selected. The import speed is high, but the graph is locked and cannot be read or written during the import. • **false**: Online import is selected. Online import is slower than offline import. However, during online import, the graph can be read (but cannot be written).
Ignore Labels on Repetitive Edges	No	Indicates whether to ignore labels on repetitive edges. The value is **Yes** or **No**, and the default value is **Yes**. • **Yes**: Indicates that the repetitive edge definition does not contain the label. That is, the <source vertex, target vertex> indicates an edge, excluding the label information. • **No**: Indicates that the repetitive edge definition contains the label. That is, the <source vertex, target vertex, label> indicates an edge.
Log Storage Path	No	Stores vertex and edge datasets that do not comply with the metadata definition, as well as detailed logs generated during graph import.

Table 6-90 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.9 MRS Kafka

Functions

The MRS Kafka node is used to query the number of messages that are not consumed by a topic.

Parameters

Table 6-91 and Table 6-92 describe the parameters of the MRS Kafka node.

Table 6-91 Parameters of MRS Kafka nodes

Parameter	Mandatory	Description
Data Connection	Yes	Select the MRS Kafka connection created in the management center.
Topic Name	Yes	Select a topic that has been created in MRS Kafka. The SDK or command line can be used to create a topic.
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).

Table 6-92 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.10 Kafka Client

Functions

The Kafka Client node is used to send data to Kafka topics.

You can learn more about how to use the Kafka Client node in **Scheduling Jobs Across Workspaces**.

Parameters

Table 6-93 describes the parameters of the Kafka Client node.

Table 6-93 Parameters of Kafka Client nodes

Parameter	Mandatory	Description
Data Connection	Yes	Select the MRS Kafka connection created in the management center.
Topic Name	Yes	Select the topic to which data is to be uploaded. If there are multiple partitions, data is sent to partition 0 by default.
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Text	Yes	Text content sent to Kafka. You can directly enter text or click  to use the EL expression.

Table 6-94 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.

Parameter	Mandatory	Description
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.11 ROMA FDI Job

Functions

The ROMA FDI Job node executes a predefined ROMA Connect data integration task to implement data integration and conversion between the source and destination.

Working Principles

This node enables you to start an FDI task or query whether an FDI task is running.

Parameters

The following table describes the parameters of a ROMA FDI Job node.

Table 6-95 Property parameters

Parameter	Mandatory	Description
Node Name	Yes	Name of the node. The value must consist of 1 to 128 characters and contain only letters, digits, and the following special characters: _-/<>
Region	Yes	Region where an existing instance resides
ROMA Instance	Yes	Select an existing ROMA instance. You can select an ROMA instance in another resource space.
FDI Task	Yes	Select an existing ROMA FDI task. You can select an FDI task in another resource space.

Table 6-96 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.

Parameter	Mandatory	Description
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.12 DLI Flink Job

Function

The DLI Flink Job node is used to create and start jobs or check whether DLI jobs are running to analyze streaming big data in real time.

After a DLI Flink streaming job is submitted to DLI, if the job is in the running state, the node is successfully executed. If periodic scheduling is configured for the job, the system periodically checks whether the Flink job is still in the running state. If the Flink job is in the running state, the node is successfully executed.

Parameters

For details about how to configure the parameters of DLI Flink jobs, see the following:

- Property parameters:

  If the job is a Flink SQL job, Flink OpenSource SQL job, or custom Flink job, the system creates and starts the job based on the job status configured on the node.

  - **Existing Flink job**: For details, see **Table 6-97**.
  - **Flink SQL job**: For details, see **Table 6-98**.
  - **Flink OpenSource SQL job**: For details, see **Table 6-99**.
  - **User-defined Flink job**: For details, see **Table 6-100**.
- Advanced parameter: **Table 6-101**

Table 6-97 Parameter parameters of an existing Flink job

Parameter	Mandatory	Description
Job Type	Yes	Select **Existing Flink job**.
Job Name	Yes	Name of an existing DLI Flink job.
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).

Table 6-98 Property parameters of a Flink SQL job

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Job Type	Yes	Select **Flink SQL job**. You can start a job by compiling SQL statements.
Job Name	Yes	Name of the DLI Flink job. It must consist of 1 to 64 characters and contain only letters, numbers, and underscores (_). The default value is the same as the node name.
Job name must be prefixed with workspace name	No	Whether to add a workspace prefix to the created job.
Script Path	Yes	Path to a Flink SQL script to be executed. If the script is not created, create and develop the Flink SQL script by referring to **Creating a Script** and **Developing an SQL Script**.

Parameter	Mandatory	Description
Script Parameter	No	If the associated Flink SQL script uses a parameter, the parameter name is displayed. Set the parameter value in the text box next to the parameter name. The parameter value can be an **EL expression**. If the parameters of the associated Flink SQL script are changed, click to refresh the parameters.
UDF Jar	No	This parameter is valid only when you select a dedicated queue for **Queue**. Before selecting a UDF JAR resource package, upload the UDF JAR package to the OBS bucket and create resources on the **Manage Resource** page. For details, see **Creating a Resource**. In SQL, you can call a user-defined function that is inserted into a JAR package.
DLI Queue	Yes	**Shared queues** are selected by default. You can also select a dedicated custom queue. **NOTE** • During job creation, a sub-user can only select a queue that has been allocated to the user. • The version of the default Spark component of the default DLI queue is not up-to-date, and an error may be reported indicating that a table creation statement cannot be executed. In this case, you are advised to create a queue to run your tasks. To enable the execution of table creation statements in the default queue, contact the customer service or technical support of the DLI service. • The default queue **default** of DLI is only used for trial. It may be occupied by multiple users at a time. Therefore, it is possible that you fail to obtain the resource for related operations. If the execution takes a long time or fails, you are advised to try again during off-peak hours or use a self-built queue to run the job.
CUs	Yes	Compute Unit (CU) is the pricing unit for DLI. A CU consists of 1 vCPU compute and 4 GB memory.
Concurrency	Yes	The number of Flink SQL jobs that run at the same time. **NOTE** The value of **Concurrency** must not exceed the value obtained through the following formula: 4 x (Number of CUs – 1).
Auto Restart upon Exception	No	Indicates whether to enable automatic restart. If this function is enabled, any job that has become abnormal will be automatically restarted.

Table 6-99 Property parameters of a Flink OpenSource SQL job

Parameters	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Job Type	Yes	Select **Flink OpenSource SQL job**. You can start a job by compiling SQL statements.
Job Name	Yes	Name of the DLI Flink job. It must consist of 1 to 64 characters and contain only letters, numbers, and underscores (_). The default value is the same as the node name.
Job name must be prefixed with workspace name	No	Whether to add a workspace prefix to the created job.
Script Path	Yes	Path to a Flink SQL script to be executed. If the script is not created, create and develop the Flink SQL script by referring to **Creating a Script** and **Developing an SQL Script**.
Script Parameter	No	If the associated Flink SQL script uses a parameter, the parameter name is displayed. Set the parameter value in the text box next to the parameter name. The parameter value can be an **EL expression**. If the parameters of the associated Flink SQL script are changed, click to refresh the parameters.
UDF Jar	No	This parameter is valid only when you select a dedicated queue for **Queue**. Before selecting a UDF JAR resource package, upload the UDF JAR package to the OBS bucket and create resources on the **Manage Resource** page. For details, see **Creating a Resource**. In SQL, you can call a user-defined function that is inserted into a JAR package.

Parameters	Mandatory	Description
DLI Queue	Yes	**Shared queues** are selected by default. You can also select a dedicated custom queue. **NOTE** • During job creation, a sub-user can only select a queue that has been allocated to the user. • The default queue **default** of DLI is only used for trial. It may be occupied by multiple users at a time. Therefore, it is possible that you fail to obtain the resource for related operations. If the execution takes a long time or fails, you are advised to try again during off-peak hours or use a self-built queue to run the job.
CUs	Yes	Compute Unit (CU) is the pricing unit for DLI. A CU consists of 1 vCPU compute and 4 GB memory.
Concurrency	Yes	The number of Flink SQL jobs that run at the same time. **NOTE** The value of **Concurrency** must not exceed the value obtained through the following formula: 4 x (Number of CUs – 1).
Auto Restart upon Exception	No	Indicates whether to enable automatic restart. If this function is enabled, any job that has become abnormal will be automatically restarted.

Table 6-100 Property parameters of a user-defined Flink job

Parameter	Mandatory	Description
Job Type	Yes	Select **User-defined Flink job**.
JAR Package	Yes	User-defined package. Before selecting a package, upload the JAR package to the OBS bucket and create resources on the **Manage Resource** page. For details, see **Creating a Resource**.
Main Class	Yes	Name of the JAR package to be loaded, for example, **KafkaMessageStreaming**. • **Default**: Specified based on the **Manifest** file in the JAR package. • **Manually assign**: Enter the class name and confirm the class arguments (separate arguments with spaces). **NOTE** When a class belongs to a package, the package path must be carried, for example, **packagePath.KafkaMessageStreaming**.

Parameter	Mandatory	Description
Main Class Parameter	Yes	List of parameters of a specified class. The parameters are separated by spaces.
DLI Queue	Yes	**Shared queues** are selected by default. You can also select a dedicated custom queue. **NOTE** • During job creation, a sub-user can only select a queue that has been allocated to the user. • The version of the default Spark component of the default DLI queue is not up-to-date, and an error may be reported indicating that a table creation statement cannot be executed. In this case, you are advised to create a queue to run your tasks. To enable the execution of table creation statements in the default queue, contact the customer service or technical support of the DLI service. • The default queue **default** of DLI is only used for trial. It may be occupied by multiple users at a time. Therefore, it is possible that you fail to obtain the resource for related operations. If the execution takes a long time or fails, you are advised to try again during off-peak hours or use a self-built queue to run the job.
Job Type	No	Select a custom image and the corresponding version. This parameter is available only when the DLI queue is a containerized queue. A custom image is a feature of DLI. You can use the Spark or Flink basic images provided by DLI to pack the dependencies (files, JAR packages, or software) required into an image using Dockerfile, generate a custom image, and release the image to SWR. Then, select the generated image and run the job. Custom images can change the container runtime environments of Spark and Flink jobs. You can embed private capabilities into custom images to enhance the functions and performance of jobs. For details about custom images, see **Overview of Custom Images**.
CUs	Yes	Compute Unit (CU) is the pricing unit for DLI. A CU consists of 1 vCPU compute and 4 GB memory.
Number of management node CUs	Yes	Set the number of CUs on a management unit. The value ranges from 1 to 4. The default value is **1**.

Parameter	Mandatory	Description
Concurrency	Yes	The number of Flink SQL jobs that run at the same time. **NOTE** The value of **Concurrency** must not exceed the value obtained through the following formula: 4 x (Number of CUs – 1).
Auto Restart upon Exception	No	Indicates whether to enable automatic restart. If this function is enabled, any job that has become abnormal will be automatically restarted.
Job Name	Yes	Name of the DLI Flink job. It must consist of 1 to 64 characters and contain only letters, numbers, and underscores (_). The default value is the same as the node name.
Job name must be prefixed with workspace name	No	Whether to add a workspace prefix to the created job.
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).

Table 6-101 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.13 DLI SQL

Functions

The DLI SQL node is used to transfer SQL statements to DLI for data source analysis and exploration.

Working Principles

This node enables you to execute DLI statements during periodical or real-time job scheduling. You can use parameter variables to perform incremental import and process partitions for your data warehouses.

Parameters

Table 6-102, **Table 6-103**, and **Table 6-104** describe the parameters of the DLI SQLnode node.

Table 6-102 Parameters of DLI SQL nodes

Parameter	Mandatory	Description
SQL Statement or Script	Yes	You can select **SQL statement** or **SQL script**. • SQL Statement Click the text box under **SQL statement** and enter the SQL statement to be executed. • SQL Script Select a script to be executed. If the script is not created, create and develop the script by repeating steps **Creating a Script** and **Developing an SQL Script**. **NOTE** If you select the SQL statement mode, the DataArts Factory module cannot parse the parameters contained in the SQL statement.

Parameter	Mandatory	Description
Database Name	Yes	Database that is configured in the SQL script. The value can be changed.
DLI Environmental Variable	No	• The environment variable must start with **dli.sql.** or **spark.sql.** • If the key of the environment variable is **dli.sql.shuffle.partitions** or **dli.sql.autoBroadcastJoinThreshold**, the environment variable cannot contain the greater than (>) or less than (<) sign. • If a parameter with the same name is configured in both a job and a script, the parameter value configured in the job will overwrite that configured in the script. **NOTE** User-defined parameter that applies to the job. Currently, the following configuration items are supported: • **dli.sql.autoBroadcastJoinThreshold**: specifies the data volume threshold to use BroadcastJoin. If the data volume exceeds the threshold, BroadcastJoin will be automatically enabled. • **dli.sql.shuffle.partitions**: specifies the number of partitions during shuffling. • **dli.sql.cbo.enabled**: specifies whether to enable the CBO optimization policy. • **dli.sql.cbo.joinReorder.enabled**: specifies whether join reordering is allowed when CBO optimization is enabled. • **dli.sql.multiLevelDir.enabled**: specifies whether to query the content in subdirectories if there are subdirectories in the specified directory of an OBS table or in the partition directory of an OBS partition table. By default, the content in subdirectories is not queried. • **dli.sql.dynamicPartitionOverwrite.enabled**: specifies that only partitions used during data query are overwritten and other partitions are not deleted.

Parameter	Mandatory	Description
Queue Name	Yes	Name of the DLI queue configured in the SQL script. The value can be changed. You can create a resource queue using either of the following methods: • Click  . On the **Queue Management** page of DLI, create a resource queue. • Go to the DLI console to create a resource queue. **NOTE** • During job creation, a sub-user can only select a queue that has been allocated to the user. • The version of the default Spark component of the default DLI queue is not up-to-date, and an error may be reported indicating that a table creation statement cannot be executed. In this case, you are advised to create a queue to run your tasks. To enable the execution of table creation statements in the default queue, contact the customer service or technical support of the DLI service. • The default queue **default** of DLI is only used for trial. It may be occupied by multiple users at a time. Therefore, it is possible that you fail to obtain the resource for related operations. If the execution takes a long time or fails, you are advised to try again during off-peak hours or use a self-built queue to run the job.
Script Parameter	No	If the associated SQL script uses a parameter, the parameter name is displayed. Set the parameter value in the text box next to the parameter name. The parameter value can be **an EL expression**. If the parameters of the associated SQL script are changed, click  to refresh the parameters.
Node Name	Yes	Name of the SQL script. The value can be changed. The rules are as follows: Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>). By default, the node name is the same as that of the selected script. If you want the node name to be different from the script name, disable this function by referring to **Disabling Auto Node Name Change**.

Parameter	Mandatory	Description
Record Dirty Data	Yes	Click to specify whether to record dirty data. • If you select , dirty data will be recorded. • If you do not select , dirty data will not be recorded. **NOTE** Dirty data refers to bad records which cannot be loaded to DLI due to incompatible data types, empty data, or incompatible data formats. If you choose to record dirty data, bad records are imported to the OBS path for storing dirty data instead of the target table. • If no OBS path for storing DLI dirty data has been configured in the workspace, the dirty data generated during DLI SQL execution is written to the **dlf-log-{projectId}** bucket by default. • To set the path for storing DLI dirty data, go to the **Workspaces** page and edit the workspace. For details, see **Configuring an OBS Bucket**.

Table 6-103 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

Table 6-104 Lineage

Parameter	Description
Input	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the output lineage.

6.11.14 DLI Spark

Functions

The DLI Spark node is used to execute a predefined Spark job.

For details about how to use the DLI Spark node, see **Developing a DLI Spark Job**.

Parameters

Table 6-105, **Table 6-106**, and **Table 6-107** describe the parameters of the DLI Sparknode node.

Table 6-105 Parameters of DLI Spark nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
DLI Queue	Yes	Select a queue from the drop-down list box. **NOTE** • During job creation, a sub-user can only select a queue that has been allocated to the user. • The version of the default Spark component of the default DLI queue is not up-to-date, and an error may be reported indicating that a table creation statement cannot be executed. In this case, you are advised to create a queue to run your tasks. To enable the execution of table creation statements in the default queue, contact the customer service or technical support of the DLI service. • The default queue **default** of DLI is only used for trial. It may be occupied by multiple users at a time. Therefore, it is possible that you fail to obtain the resource for related operations. If the execution takes a long time or fails, you are advised to try again during off-peak hours or use a self-built queue to run the job.
Spark Version	No	Select the version of the Spark component. If there is no specific requirement on the version, use the default version 2.3.2.

Parameter	Mandatory	Description
Job Type	No	Type of the Spark image used by the job. The following options are available: **Basic**, **AI-enhanced**, and **Image**. If you select **Image**, you need to set the image name and version. This parameter is available only when the DLI queue is a containerized queue. A custom image is a feature of DLI. You can use the Spark or Flink basic images provided by DLI to pack the dependencies (files, JAR packages, or software) required into an image using Dockerfile, generate a custom image, and release the image to SWR. Then, select the generated image and run the job. Custom images can change the container runtime environments of Spark and Flink jobs. You can embed private capabilities into custom images to enhance the functions and performance of jobs.
Job Name	Yes	Name of the DLI Spark job. The name must contain 1 to 64 characters, including only letters, numbers, and underscores (_). The default value is the same as the node name.
Job Running Resources	No	Select the running resource specifications of the job. ● 8-core, 32 GB memory ● 16-core, 64 GB memory ● 32-core, 128 GB memory
Major Job Class	Yes	Name of the major class of the Spark job. When the application type is **.jar**, the main class name cannot be empty.
Spark program resource package	Yes	JAR file on which the Spark job depends. You can enter the JAR package name or the corresponding OBS path. The format is as follows: **obs://*Bucket name/Folder name/Package name***. Before selecting a resource package, upload the JAR package and its dependency packages to the OBS bucket and create resources on the **Manage Resource** page. For details, see **Creating a Resource**.

Parameter	Mandatory	Description
Resource Type	Yes	Select **OBS path** or **DLI program package**. • **OBS path**: The resource package file will not be uploaded to DLI resource management system before the job is executed. The OBS path where the file is located is part of the message body for starting the job. This type is recommended. • DLI package: The resource package file will be uploaded to the DLI resource management system before the job is executed.
Group	No	This parameter is mandatory when **Resource Type** is set to **DLI program package**. You can select **Use existing**, **Create new**, or **Do not use**.
Group Name	No	This parameter is mandatory when **Resource Type** is set to **DLI program package**. • **Use existing**: Select an existing group. • **Create new**: Enter a user-defined group name. • **Do not use**: Do not select or enter a group name.
Major-Class Entry Parameters	No	User-defined parameters. Separate multiple parameters by **Enter**. These parameters can be replaced by global variables. For example, if you create a global variable **batch_num** on the **Global Configuration** > **Global Variables** page, you can use **{{batch_num}}** to replace a parameter with this variable after the job is submitted.
Spark Job Running Parameters	No	Enter a parameter in the format of **key/value**. Press Enter to separate multiple key-value pairs. For details about the parameters, see **Spark Configuration**. These parameters can be replaced by global variables. For example, if you create a global variable **custom_class** on the **Global Configuration** > **Global Variables** page, you can use **"spark.sql.catalog"={{custom_class}}** to replace a parameter with this variable after the job is submitted. **NOTE** The JVM garbage collection algorithm cannot be customized for Spark jobs.

Parameter	Mandatory	Description
Module Name	No	Dependency modules provided by DLI for executing datasource connection jobs. To access different services, you need to select different modules. • CloudTable/MRS HBase: sys.datasource.hbase • DDS: sys.datasource.mongo • CloudTable/MRS OpenTSDB: sys.datasource.opentsdb • DWS: sys.datasource.dws • RDS MySQL: sys.datasource.rds • RDS PostGre: sys.datasource.rds • DCS: sys.datasource.redis • CSS: sys.datasource.css DLI internal modules include: • sys.res.dli-v2 • sys.res.dli • sys.datasource.dli-inner-table
Metadata Access	Yes	Whether to access metadata through Spark jobs. For details, see **Using the Spark Job to Access DLI Metadata**.

Table 6-106 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

Table 6-107 Lineage

Parameter	Description
Input	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the output lineage.

6.11.15 DWS SQL

Functions

The DWS SQL node is used to transfer SQL statements to DWS.

For details about how to use the DWS SQL operator, see **Developing a DWS SQL Script and Job**.

Context

This node enables you to execute DWS statements during batch or real-time job processing. You can use parameter variables to perform incremental import and process partitions for your data warehouses.

Parameters

Table 6-108, **Table 6-109**, and **Table 6-110** describe the parameters of the DWS SQLnode node.

Table 6-108 Parameters of DWS SQL nodes

Parameter	Mandatory	Description
SQL or Script	Yes	You can select **SQL statement** or **SQL script**. • SQL Statement Click the text box under **SQL statement** and enter the SQL statement to be executed. • SQL Script Select a script to be executed. If the script is not created, create and develop the script by repeating steps **Creating a Script** and **Developing an SQL Script**. **NOTE** If you select the SQL statement mode, the DataArts Factory module cannot parse the parameters contained in the SQL statement.
Data Connection	Yes	Data connection that is configured in the SQL script. The value can be changed.
Database	Yes	Database that is configured in the SQL script. The value can be changed.
Script Parameter	No	If the associated SQL script uses a parameter, the parameter name is displayed. Set the parameter value in the text box next to the parameter name. The parameter value can be **an EL expression**. If the parameters of the associated SQL script are changed, click to refresh the parameters.

Parameter	Mandatory	Description
Dirty Data Table	No	Name of the dirty data table defined in the SQL script. The dirty data attributes cannot be edited. They are automatically recommended by the SQL script content. Syntax for the DWS dirty data table: **with** table_name or **log into** table_name
Matching Rule	N/A	Enter a Java regular expression used to match the DWS SQL result. For example, if the expression is (?<=\()(-*\d+?)(?=,) and the SQL result is (1,"error message"), then the matched result is "1".
Failure Matching Value	N/A	If the matched content equals the set value, the node fails to be executed.
Node Name	Yes	Name of the SQL script. The value can be changed. The rules are as follows: Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>). By default, the node name is the same as that of the selected script. If you want the node name to be different from the script name, disable this function by referring to **Disabling Auto Node Name Change**.

Table 6-109 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

Table 6-110 Lineage

Parameter	Description
Input	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the output lineage.

6.11.16 MRS Spark SQL

Functions

The MRS Spark SQL node is used to execute a predefined SparkSQL statement on MRS.

Parameters

Table 6-111, **Table 6-112**, and **Table 6-113** describe the parameters of the MRS Spark SQLnode node.

Table 6-111 Parameters of MRS Spark SQL nodes

Parameter	Mandatory	Description
MRS Job Name	No	MRS job name. If the MRS job name is not set and the direct connection mode is selected, the node name can contain a maximum of 64 characters and can only consist of letters, digits, hyphens (-), and underscores (_). The system can automatically enter an MRS job name in *Job name_Node name* format.
SQL Script	Yes	Path of a script to be executed. If no script is available, create and develop a script by referring to **Creating a Script** and **Developing an SQL Script**.
Data Connection	Yes	Data connection that is configured in the SQL script. The value can be changed.
MRS Resource Queue	No	Select a created MRS resource queue. **NOTE** • If you have selected an MRS API connection, you can configure resources (such as threads, memory, CPUs, and MRS resource queues) specially for the Spark SQL job. However, you cannot do so if you have selected a proxy connection. • Select a queue you configured in the queue permissions of DataArts Security. If you set multiple resource queues for this node, the resource queue you select here has the highest priority.
Database	Yes	Database that is configured in the SQL script. The value can be changed.

Parameter	Mandatory	Description
Script Parameter	No	If the associated SQL script uses a parameter, the parameter name is displayed. Set the parameter value in the text box next to the parameter name. The parameter value can be **an EL expression**. If the parameters of the associated SQL script are changed, click to refresh the parameters.
Program Parameter	No	Used to configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. **NOTE** • If you have selected an MRS API connection, you can configure resources (such as threads, memory, CPUs, and MRS resource queues) specially for the Spark SQL job. However, you cannot do so if you have selected a proxy connection. • This parameter is mandatory if the cluster version is MRS 1.8.7 or later than MRS 2.0.1. For details about the program parameters of MRS SparkSQL jobs, see **Running a SparkSql Job** > **Table 2 Program Parameter parameters** in the *MapReduce User Guide*.
Node Name	Yes	Name of the SQL script. The value can be changed. Node name. It contains 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. **NOTE** The node name cannot contain Chinese characters or more than 64 characters. If the node name does not meet requirements, the MRS job will fail to be submitted. By default, the node name is the same as that of the selected script. If you want the node name to be different from the script name, disable this function by referring to **Disabling Auto Node Name Change**.

Table 6-112 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.

Parameter	Mandatory	Description
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

Table 6-113 Lineage

Parameter	Description
Input	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.

Parameter	Description
View Details	Click to view details about the table created based on the output lineage.

6.11.17 MRS Hive SQL

Functions

The MRS Hive SQL node is used to execute a predefined Hive SQL script in DataArts Factory.

For details about how to use the MRS Hive SQL node, see **Developing a Hive SQL Job**.

NOTE

MRS Hive SQL nodes do not support Hive transaction tables.

Parameters

Table 6-114, **Table 6-115**, and **Table 6-116** describe the parameters of the MRS Hive SQLnode node.

Table 6-114 Parameters of MRS Hive SQL nodes

Parameter	Mandatory	Description
MRS Job Name	No	MRS job name. If the MRS job name is not set and the direct connection mode is selected, the node name can contain a maximum of 64 characters and can only consist of letters, digits, hyphens (-), and underscores (_). The system can automatically enter an MRS job name in *Job name_Node name* format.
SQL Script	Yes	Path of a script to be executed. If no script is available, create and develop a script by referring to **Creating a Script** and **Developing an SQL Script**.
Data Connection	Yes	Data connection that is configured in the SQL script. The value can be changed.
Database	Yes	Database that is configured in the SQL script. The value can be changed.

Parameter	Mandatory	Description
MRS Resource Queue	No	Select a created MRS resource queue. **NOTE** Select a queue you configured in the queue permissions of DataArts Security. If you set multiple resource queues for this node, the resource queue you select here has the highest priority.
Script Parameter	No	If the associated SQL script uses a parameter, the parameter name is displayed. Set the parameter value in the text box next to the parameter name. The parameter value can be **an EL expression**. If the parameters of the associated SQL script are changed, click to refresh the parameters.
Program Parameter	No	Used to configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. **NOTE** This parameter is mandatory if the cluster version is MRS 1.8.7 or later than MRS 2.0.1. For details about the program parameters of MRS Hive SQL jobs, see **Running a HiveSql Job** > **Table 2 Program Parameter parameters** in the *MapReduce Service User Guide*.
Node Name	Yes	Name of the SQL script. The value can be changed. The rules are as follows: Node name. It contains 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. **NOTE** The node name cannot contain Chinese characters or more than 64 characters. If the node name does not meet requirements, the MRS job will fail to be submitted. By default, the node name is the same as that of the selected script. If you want the node name to be different from the script name, disable this function by referring to **Disabling Auto Node Name Change**.

Table 6-115 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.

Parameter	Mandatory	Description
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

Table 6-116 Lineage

Parameter	Description
Input	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click  to modify the parameter settings. After the modification, save the settings.

Parameter	Description
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the output lineage.

6.11.18 MRS Presto SQL

Functions

The MRS Presto SQL node is used to execute the Presto SQL script predefined in DataArts Factory.

Parameters

Table 6-117, **Table 6-118**, and **Table 6-119** describe the parameters of the MRS Presto SQL node.

Table 6-117 Property parameters

Parameters	Mandatory	Description
SQL or Script	Yes	You can select **SQL statement** or **SQL script**. • SQL Statement Click the text box under **SQL statement** and enter the SQL statement to be executed. • SQL Script Select a script to be executed. If the script is not created, create and develop the script by repeating steps **Creating a Script** and **Developing an SQL Script**. **NOTE** If you select the SQL statement mode, the DataArts Factory module cannot parse the parameters contained in the SQL statement.
Data Connection	Yes	Data connection that is configured in the SQL script. The value can be changed.
Schema	Yes	Database that is configured in the SQL script. The value can be changed.
Script Parameter	No	If the associated SQL script uses a parameter, the parameter name is displayed. Set the parameter value in the text box next to the parameter name. The parameter value can be **an EL expression**. If the parameters of the associated SQL script are changed, click to refresh the parameters.
Node Name	Yes	Name of the SQL script. The value can be changed. Node name. It contains 1 to 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. **NOTE** The node name cannot contain Chinese characters or more than 64 characters. If the node name does not meet requirements, the MRS job will fail to be submitted. By default, the node name is the same as that of the selected script. If you want the node name to be different from the script name, disable this function by referring to **Disabling Auto Node Name Change**.

Table 6-118 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.

Parameter	Mandatory	Description
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

Table 6-119 Lineage

Parameter	Description
Input	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click  to modify the parameter settings. After the modification, save the settings.

Parameter	Description
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the output lineage.

6.11.19 MRS Spark

Functions

The MRS Spark node is used to execute a predefined Spark job on MRS.

Parameters

Table 6-120, Table 6-121, and Table 6-122 describe the parameters of the MRS Sparknode node.

Table 6-120 Parameters of MRS Spark nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>). By default, the node name is the same as that of the selected script. If you want the node name to be different from the script name, disable this function by referring to **Disabling Auto Node Name Change**.

Parameter	Mandatory	Description
MRS Cluster Name	Yes	Name of the MRS cluster. To create an MRS cluster, use either of the following methods: • Click  . On the **Clusters** page, create an MRS cluster. • Go to the MRS console to create an MRS cluster.
MRS Resource Queue	No	Select a created MRS resource queue. **NOTE** Select a queue you configured in the queue permissions of DataArts Security. If you set multiple resource queues for this node, the resource queue you select here has the highest priority.
Spark Job Name	Yes	MRS job name. It can contain a maximum of 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. The system can automatically enter a job name in *Job name_Node name* format. **NOTE** The job name cannot contain Chinese characters or more than 64 characters. If the job name does not meet requirements, the MRS job will fail to be submitted.
Process Type	Yes	Processing type of the Spark job • **Batch**: The node waits for the Spark job execution to complete. • **Stream**: The node is executed as long as the job is successfully started. Each time the job is scheduled in the future, the system checks whether the job is in running state. If the job is in running state, it is successfully executed. Note that this parameter only specifies the processing mode. You must set parameters for the selected mode.
JAR Package	Yes	Select a JAR package. Before selecting a JAR package, upload the JAR package to the OBS bucket, create a resource on the **Manage Resource** page, and add the JAR package to the resource management list. For details, see **Creating a Resource**.
JAR File Parameters	No	Parameters of the JAR package.

Parameter	Mandatory	Description
Program Parameter	No	Used to configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. **NOTE** This parameter is mandatory if the cluster version is MRS 1.8.7 or later than MRS 2.0.1. For details about the program parameters of MRS Spark jobs, see **Running a Spark Job** in the *MapReduce Service User Guide*.
Input Data Path	No	Path where the input data resides.
Output Data Path	No	Path where the output data resides.

Table 6-121 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

Table 6-122 Lineage

Parameter	Description
Input	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the output lineage.

6.11.20 MRS Spark Python

Functions

The MRS Spark Python node is used to execute a predefined Spark Python job on MRS.

For details about how to use the MRS Spark Python operator, see **Developing an MRS Spark Python Job**.

Parameters

Table 6-123, **Table 6-124**, and **Table 6-125** describe the parameters of the MRS Spark Pythonnode node.

Table 6-123 Parameters of MRS Spark Python nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Job Name	Yes	MRS job name. It can contain a maximum of 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. The system can automatically enter a job name in *Job name_Node name* format. **NOTE** The job name cannot contain Chinese characters or more than 64 characters. If the job name does not meet requirements, the MRS job will fail to be submitted.
Script Type	Yes	• Offline • Online
MRS Cluster Name	Yes	Select an MRS cluster that supports Spark Python. Only a specific version of MRS supports Spark Python. Test the cluster first to ensure that it supports Spark Python. To create an MRS cluster, use either of the following methods: • Click . On the **Clusters** page, create an MRS cluster. • Go to the MRS console to create an MRS cluster. For details about how to create a cluster, see **Buying a Custom Cluster** in *MapReduce Service (MRS) Usage Guide*.

Parameter	Mandatory	Description
MRS Resource Queue	No	Select a created MRS resource queue. **NOTE** Select a queue you configured in the queue permissions of DataArts Security. If you set multiple resource queues for this node, the resource queue you select here has the highest priority.
SQL Script	Yes	This parameter is available only when **Script Type** is set to **Online**. Select a Spark Python script.
Script Parameter	No	This parameter is available only when **Script Type** is set to **Online**. If the associated Spark Python script uses a parameter, the parameter name is displayed. Set the parameter value in the text box next to the parameter name.
Program Parameter	No	This parameter is available only when **Script Type** is set to **Online**. Configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. **NOTE** This parameter is mandatory if the cluster version is MRS 1.8.7 or later than MRS 2.0.1. For details about the program parameters of MRS Spark jobs, see **Running a Spark Job** in the *MapReduce Service User Guide*.
Parameter	Yes	This parameter is available only when **Script Type** is set to **Offline**. Enter parameters. Press **Enter** between parameters.
Execution Program Parameter	No	Enter parameters of the MRS execution program. Use spaces to separate parameters. To prevent parameters from being saved as plaintext, add an at sign (@) before parameters.
Attribute	No	Enter parameters in the key=value format. Use **Enter** to separate multiple parameters.

Table 6-124 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.

Parameter	Mandatory	Description
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

Table 6-125 Lineage

Parameter	Description
Input	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.

Parameter	Description
View Details	Click to view details about the table created based on the output lineage.

6.11.21 MRS ClickHouse

Functions

The MRS ClickHouse node is used to execute the ClickHouse SQL script predefined in DataArts Factory.

Parameters

Table 6-126, **Table 6-127**, and **Table 6-128** describe the parameters of the MRS ClickHouse node.

Table 6-126 Parameters of MRS ClickHouse nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>). By default, the node name is the same as that of the selected script. If you want the node name to be different from the script name, disable this function by referring to **Disabling Auto Node Name Change**.
SQL Script	Yes	Path of a script to be executed. If no script is available, create and develop a script by referring to **Creating a Script** and **Developing an SQL Script**.
Script Parameter	No	If the associated SQL script uses a parameter, the parameter name is displayed. Set the parameter value in the text box next to the parameter name. The parameter value can be an **EL expression**. If the parameters of the associated SQL script are changed, click to refresh the parameters.
Data Connection	Yes	Data connection that is configured in the SQL script. The value can be changed.
Database	Yes	Database that is configured in the SQL script. The value can be changed.

Parameter	Mandatory	Description
Program Parameter	No	Used to configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. **NOTE** This parameter is mandatory if the cluster version is MRS 1.8.7 or later than MRS 2.0.1.

Table 6-127 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.

Parameter	Mandatory	Description
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

Table 6-128 Lineage

Parameter	Description
Input	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click  to modify the parameter settings. After the modification, save the settings.

Parameter	Description
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the output lineage.

6.11.22 MRS Impala SQL

Functions

The MRS Impala SQL node is used to execute the Impala SQL script predefined in DataArts Factory.

Parameters

Table 6-129 and Table 6-130 describe the parameters of the MRS Impala node.

Table 6-129 Parameters

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>). By default, the node name is the same as that of the selected script. If you want the node name to be different from the script name, disable this function by referring to **Disabling Auto Node Name Change**.

Parameter	Mandatory	Description
SQL Script	Yes	Path of a script to be executed. If no script is available, create and develop a script by referring to **Creating a Script** and **Developing an SQL Script**.
Data Connection	Yes	Data connection that is configured in the SQL script. The value can be changed.
Database	Yes	Database that is configured in the SQL script. The value can be changed.
Script Parameter	No	If the associated SQL script uses a parameter, the parameter name is displayed. Set the parameter value in the text box next to the parameter name. The parameter value can be **an EL expression**. If the parameters of the associated SQL script are changed, click to refresh the parameters.

Table 6-130 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

Table 6-131 Lineage

Parameter	Description
Input	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the output lineage.

6.11.23 MRS Flink Job

Functions

The MRS Flink Job node is used to execute the Flink SQL script and Flink job predefined in DataArts Factory.

For details about how to use the MRS Flink Job node, see **Developing an MRS Flink Job**.

Parameters

Table 6-132 and **Table 6-133** describe the parameters of the MRS Flink node.

Table 6-132 Parameters of the MRS Flink node

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Job Type	Yes	The following options are available: • Flink SQL job • User-defined Flink job
Script Path	Yes	This parameter is available when you select **Flink SQL job** for **Job Type**. Select the Flink SQL script to be executed. If no Flink SQL script is available, create and develop one by referring to **Creating a Script** and **Developing an SQL Script**.
Script Parameter	No	This parameter is available when you select **Flink SQL job** for **Job Type**. If the associated SQL script uses a parameter, the parameter name is displayed. Set the parameter value in the text box next to the parameter name. The parameter value can be **an EL expression**. If the parameters of the associated SQL script are changed, click to refresh the parameters.

Parameter	Mandatory	Description
Process Type	Yes	Set the mode of the Flink job. • **Batch**: The node waits for the Flink job execution to complete. • **Stream**: The node is executed as long as the job is successfully started. Each time the job is scheduled in the future, the system checks whether the job is in running state. If the job is in running state, it is successfully executed. Note that this parameter only specifies the processing mode. You must set parameters for the selected mode.
MRS Cluster Name	Yes	Select an MRS cluster. To create an MRS cluster, use either of the following methods: • Click . On the **Clusters** page, create an MRS cluster. • Go to the MRS console to create an MRS cluster. **NOTE** Currently, MRS Flink jobs support MRS 3.2.0-LTS.1 and later versions.
Job Name	Yes	MRS job name. It can contain a maximum of 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. The system can automatically enter a job name in *Job name_Node name* format. **NOTE** The job name cannot contain Chinese characters or more than 64 characters. If the job name does not meet requirements, the MRS job will fail to be submitted.
Job Resource Package	Yes	Select a JAR package. Before selecting a JAR package, upload the JAR package to the OBS bucket, create a resource on the **Manage Resource** page, and add the JAR package to the resource management list. For details, see **Creating a Resource**.
Job Execution Parameter	No	Key parameter of the program that executes the Flink job. This parameter is specified by a function in the user program. Multiple parameters are separated by space.
MRS Resource Queue	No	Select a created MRS resource queue. **NOTE** Select a queue you configured in the queue permissions of DataArts Security. If you set multiple resource queues for this node, the resource queue you select here has the highest priority.

Parameter	Mandatory	Description
Program Parameter	No	Used to configure optimization parameters such as threads, memory, and vCPUs for the job to optimize resource usage and improve job execution performance. **NOTE** This parameter is mandatory if the cluster version is MRS 1.8.7 or later than MRS 2.0.1. For details about the program parameters of MRS Spark jobs, see **Running a Flink Job** in the *MapReduce Service User Guide*.
Input Data Path	No	Path where the input data resides.
Output Data Path	No	Path where the output data resides.

Table 6-133 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.

Parameter	Mandatory	Description
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.24 MRS MapReduce

Functions

The MRS MapReduce node is used to execute a predefined MapReduce program on MRS.

Parameters

Table 6-134 and **Table 6-135** describe the parameters of the MRS MapReduce node.

Table 6-134 Parameters of MRS MapReduce nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
MRS Cluster Name	Yes	Name of the MRS cluster. To create an MRS cluster, use either of the following methods: • Click . On the **Clusters** page, create an MRS cluster. • Go to the MRS console to create an MRS cluster.
MapReduce Job Name	Yes	MRS job name. It can contain a maximum of 64 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. **NOTE** The job name cannot contain Chinese characters or more than 64 characters. If the job name does not meet requirements, the MRS job will fail to be submitted.
JAR Package	Yes	Select a JAR package. Before selecting a JAR package, upload the JAR package to the OBS bucket, create a resource on the **Manage Resource** page, and add the JAR package to the resource management list. For details, see **Creating a Resource**.
JAR File Parameters	No	Parameters of the JAR package.
Input Data Path	No	Path where the input data resides.
Output Data Path	No	Path where the output data resides.

Table 6-135 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.25 CSS

Functions

The CSS node is used to process CSS requests and enable online distributed searching.

Parameters

Table 6-136 and Table 6-137 describe the parameters of the CSS node.

Table 6-136 Parameters of CSS nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Cluster or Data Connection	Yes	Select **Cluster** or **Connection**. If you want to enable the security mode for CSS clusters, select **Connection**.
CloudSearch Cluster	Yes	This parameter is required when you select **Cluster** for **Cluster or Data Connection**. Connection to CloudSearch. A CloudSearch cluster has been created in CloudService. Currently, only clusters of version 5.5.1 is supported.
CDM Cluster Name	Yes	This parameter is required when you select **Cluster** for **Cluster or Data Connection**. Name of the selected CDM cluster. The CDM cluster functions as a proxy to forward requests. If there are no CDM clusters available in the drop-down list, create one on the CDM console.

Parameter	Mandatory	Description
Data Connection	Yes	This parameter is required when you select **Connection** for **Cluster or Data Connection**. Select a data connection.
Request Type	Yes	Possible values: • **GET** • **POST** • **PUT** • **HEAD** • **DELETE**
Request Parameter	No	Parameter of the request. For example, to query the dlfdata mapping type in the dlf_search index, set this parameter to: **/dlf_search/dlfdata/_search**
Request Body	No	The request body is in JSON format. This parameter is mandatory when **Request Type** is **POST**, **PUT**, or **HEAD**.
CloudSearch Output Path	No	Path where output data is to be stored.

Table 6-137 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.26 Shell

Functions

The Shell node is used to execute a shell script.

NOTE

With EL expression **#{Job.getNodeOutput()}**, you can obtain the desired content (4000 characters at most and counted backwards) in the output of the shell script run by the Shell node.

Example:

To obtain **<name>jack<name1>** from a shell script (script name: shell_job1) output, enter the following EL expression:

```
#{StringUtil.substringBetween(Job.getNodeOutput("shell_job1"),"<name>","<name1>")}
```

Parameters

Table 6-138 and **Table 6-139** describe the parameters of the Shell node.

Table 6-138 Parameters

Parameter	Mandatory	Description
Shell or Script	Yes	You can select **Shell statement** or **Shell script**. • Shell statement In the **Shell statement** text box, enter the Shell statement to be executed. • Shell script Select a script to be executed. If no script is available, create and develop a script by referring to **Creating a Script** and **Developing a Shell Script**. **NOTE** If you select **Shell statement**, the DataArts Factory module cannot parse the parameters contained in the Shell statement. The execution result of a Shell node cannot be larger than 30 MB. Otherwise, an error is reported.
Host Connection	Yes	Selects the host where a shell script is to be executed. **NOTICE** • The maximum number of shell or Python scripts that can run concurrently on the ECS is determined by the value of **MaxSessions** in the **/etc/ssh/sshd_config** file on the ECS. Set **MaxSessions** based on the scheduling frequency of shell or Python scripts. • You have the permission to create and execute files in the **/tmp** directory on the host. • Shell and Python scripts are executed in the **/tmp** directory on an ECS. Ensure that the disk space of the **/tmp** directory is not used up.
Script Parameter	No	Parameter transferred to the script when the shell script is executed. Parameters are separated by spaces. For example: **a b c**. The parameter must be referenced by the shell script. Otherwise, the parameter is invalid.
Interactive Input	No	Interactive information (for example, passwords) provided during shell script execution. Interactive parameters are separated by spaces. The shell script reads parameter values in sequence according to the interaction situation. The following is an example of using the read -p syntax: `read -p "Parameter 1 and parameter 2"Variable 1 Variable 2`

Parameter	Mandatory	Description
Node Name	Yes	Name of the node. It contains a maximum of 128 characters, including letters, digits, hyphens (-), underscores (_), slashes (/), angle brackets (<>), and periods (.). By default, the node name is the same as that of the selected script. If you want the node name to be different from the script name, disable this function by referring to **Disabling Auto Node Name Change**.

Table 6-139 Advanced settings

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system checks whether the node execution is complete. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.

Parameter	Mandatory	Description
Retry Condition	No	If **Retry upon Failure** is set to **Yes**, retry conditions can be set. Enable **Retry Condition** and set the return code range. The shell job can determine whether to retry a failed node based on the return code. You can define the return codes that can be used to determine whether to retry a failed node.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: Execution of the current job will stop, and the job instance status will become **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select **Dry run**, the node will not be executed, and a success message will be returned.

6.11.27 RDS SQL

Functions

The RDS SQL node is used to transfer SQL statements to RDS.

Parameters

Table 6-140 and **Table 6-141** describe the parameters of the RDS SQL node.

Table 6-140 Parameters of RDS SQL nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>). By default, the node name is the same as that of the selected script. If you want the node name to be different from the script name, disable this function by referring to **Disabling Auto Node Name Change**.
Data Connection	Yes	Name of the data connection.
Database	Yes	Name of the database. The database has been created. You are advised not to use the default database.
SQL or Script	Yes	You can select **SQL statement** or **SQL script**. • SQL statement Click the text box under **SQL statement** and enter the SQL statement to be executed. • SQL script Select a script to be executed. If no script is available, create and develop a script by referring to **Creating a Script** and **Developing an SQL Script**. **NOTE** If you select the SQL statement mode, the DataArts Factory module cannot parse the parameters contained in the SQL statement.

Table 6-141 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.28 ETL Job

Functions

The ETL Job node is used to extract data from a specified data source, preprocess the data, and import the data to the target data source.

NOTE

The destination is the ETL Job node of DWS and does not support scheduling using an agency. You are advised to use a public IAM account with better compatibility for scheduling. For details, see **Configuring a Scheduling Identity**.

Parameters

Table 6-142, **Table 6-143**, and **Table 6-144** describe the parameters of the ETL Job node.

Table 6-142 Parameters of Transform Load nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).

Parameter	Mandatory	Description
ETL Configuration	Yes	Click  to edit the source and destination data to be transformed. The supported source data types are DLI, OBS and MySQL. ● When the source data type is DLI, the supported destination data types are DWS, GES, CSS, OBS, and DLI. ● When the source data type is MySQL, the supported destination data type is MySQL. ● When the source data type is OBS, the supported destination data can be of the DLI type and the DWS type. **NOTICE** ● Data transformation from DLI to DWS: Before importing data from DataArts Factory to DWS, ensure that a DWS data connection and a table have been created. Before importing data from DLI to DWS, ensure that a DWS table have been created. ● Data transformation from DLI to CSS: Before importing data from DLI to CSS, ensure that a cross-source connection associated with CSS has been created on DLI. For details about how to create a cross-source connection on DLI, see *Data Lake Insight User Guide*.
Configure SQL Template	No	Click **Obtain Template** to obtain an SQL template.

Table 6-143 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

Table 6-144 Lineage

Parameter	Description
Input	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the output lineage.

6.11.29 Python

NOTICE

Before using a Python node, ensure that the host connected to the node has an environment for executing Python scripts.

Functions

The Python node is used to execute Python statements.

For details about how to use the Python node, see **Developing a Python Script**.

NOTE

Python nodes support script parameters and job parameters.

Parameters

Table 6-145 and **Table 6-146** describe the parameters of the Python node.

Table 6-145 Parameters of the Python node

Parameter	Mandatory	Description
Python Statement or Script	Yes	You can select **Python statement** or **Python script**. • Python statement Click the text box under **Python Statement**. In the displayed **Python Statement** dialog box, enter the Python statement to be executed. • Python script In **Python Script**, select the Python script to be executed. The Python version is displayed by default, for example, **Python3**. If no script is available, create and develop a script by referring to **Creating a Script** and **Developing a Python Script**. **NOTE** • If you select **Python statement**, the DataArts Factory module cannot parse the parameters contained in the Python statement. • If you select **Python script**, the system displays the Python version selected during Python script creation by default. • For existing jobs, **Python2** is used by default. • The execution result of a Python node cannot be larger than 30 MB. Otherwise, an error is reported.

Parameter	Mandatory	Description
Host Connection	Yes	Select the host where the Python statement is to be executed. Ensure that the host has an environment for executing Python scripts. **NOTICE** • The maximum number of shell or Python scripts that can run concurrently on the ECS is determined by the value of **MaxSessions** in the **/etc/ssh/sshd_config** file on the ECS. Set **MaxSessions** based on the scheduling frequency of shell or Python scripts. • You have the permission to create and execute files in the **/tmp** directory on the host. • Shell and Python scripts are executed in the **/tmp** directory on an ECS. Ensure that the disk space of the **/tmp** directory is not used up.
Script Parameters	No	Parameter transferred to the script when the Python statement is executed. Parameters are separated by spaces. For example: **a b c**. The parameter must be referenced by the Python statement. Otherwise, the parameter is invalid.
Interactive Input	No	Interactive information (passwords, for example) provided during Python statement execution. Interactive parameters are separated by spaces. The Python statement reads parameter values in sequence according to the interaction situation.
Node Name	Yes	Name of the node. The value must consist of 1 to 128 characters and contain only letters, digits, and the following special characters: _-/<>. By default, the node name is the same as that of the selected script. If you want the node name to be different from the script name, disable this function by referring to **Disabling Auto Node Name Change**.

Table 6-146 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.30 ModelArts Train

Function

You can orchestrate ModelArts Train operators to schedule the ModelArts workflow in DataArts Studio.

Parameters

Table 6-147 and Table 6-148 describe the parameters of the ModelArts Train node.

Table 6-147 Parameters of the ModelArts Train node

Parameter	Mandatory	Description
ModelArts Workspace	Yes	ModelArts workspace. The workspace must be in the same region as DataArts Studio.
Workflow Version	Yes	ModelArts workflow version • V1 • V2
ModelArts Workflow	Yes	ModelArts workflow. The workflow must be in the same region as DataArts Studio.
Node Name	Yes	Name of the node. The value must consist of 1 to 128 characters and contain only letters, digits, and the following special characters: _-/<>.

Table 6-148 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.

Parameter	Mandatory	Description
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.31 Http Trigger

Functions

Http Trigger is a cross-platform scheduling trigger node of DataArts Studio. If you want to trigger a job on DataArts Studio after a job on another scheduling system is complete, you can use the Http Trigger node.

Parameters

Table 6-149 describes the parameter of the Http Trigger node.

Table 6-149 Parameter of the Http Trigger node

Parameter	Mandatory	Description
Node ID	No	ID of the Http Trigger node. When you add an Http Trigger node, the system automatically generates a node ID, which is unique in the workspace and cannot be changed.
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Maximum Wait Time	Yes	Maximum wait time for HTTP messages. If no message is received within this time, the node and its subsequent nodes are canceled and the job is stopped. The value ranges from 1 to 24 hours. The default value is 24 hours.

6.11.32 Create OBS

NOTE

The OBS path cannot be a log path starting with **s3a://**.

Constraints

This function depends on OBS.

Functions

The Create OBS node is used to create buckets and directories on OBS.

Parameters

Table 6-150 and **Table 6-151** describe the parameters of the Create OBS node.

Table 6-150 Parameters of Create OBS nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
OBS Path	Yes	Path to the OBS bucket or directory. • To create a bucket, enter **//***OBS bucket name*. The OBS bucket name must be unique • To create an OBS directory, select the path to the OBS directory to be created, and enter the /*Directory name* following the path. The directory name must be unique.

Table 6-151 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.33 Delete OBS

Constraints

This function depends on OBS.

Functions

The Delete OBS node is used to delete a bucket or directory on OBS.

Parameters

Table 6-152 and Table 6-153 describe the parameters of the Delete OBS node.

Table 6-152 Parameters of Delete OBS nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
OBS Path	Yes	Path to the OBS bucket or directory. **NOTE** If you delete an OBS bucket or directory, files stored in it are also deleted and cannot be restored. Before you delete a bucket or directory, back up the files stored in it if they need to be retained.

Table 6-153 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.

Parameter	Mandatory	Description
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.34 OBS Manager

Constraints

This function depends on OBS.

Function

The OBS Manager node is used to move or copy files from an OBS bucket to a specified directory.

Parameters

Table 6-154, **Table 6-155**, and **Table 6-156** describe the parameters of the OBS Managernode node.

Table 6-154 Parameters of OBS Manager nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).

Parameter	Mandatory	Description
Operation Type	Yes	Operations that can be performed on the node. • **Move File**: moves a source file or directory to a new directory. • **Copy File**: copies the source file or directory. • **Rename File**: renames the last level of the directory or file. For example, you can rename the directory **obs://test/a/b/c/** as **obs://test/a/b/d/**, and rename the file **obs://test/a/b/hello.txt** as **obs://test/a/b/bye.txt**. • **Monitor File**: checks whether a file or directory exists. If the file or directory exists, the node is executed successfully. Otherwise, the node fails to be executed. If you want the job to be handled in different ways based on whether the file or directory exists, you can set an IF condition based on the execution status of the node. For details, see **IF Statements**.
Source File or Directory	Yes	OBS file or directory to be managed in the OBS bucket.
Target Directory	Yes	Directory for storing OBS files to be moved or copied from the OBS bucket.
File Filter	No	Wildcard for file filtering. Only the files that meet the filtering condition can be moved or copied. If this parameter is not specified, all source files are moved by default. For example, when you enter *.csv, files in this format will be moved or copied.

Table 6-155 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

Table 6-156 Lineage

Parameter	Description
Input	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the input lineage.
Output	
Add	Click **Add**. In the **Type** drop-down list, select the type to be created. The value can be **DWS**, **OBS**, **CSS**, **HIVE**, **DLI**, or **CUSTOM**.
OK	Click **OK** to save the parameter settings.
Cancel	Click **Cancel** to cancel the parameter settings.
Modify	Click to modify the parameter settings. After the modification, save the settings.
Delete	Click to delete the parameter settings.
View Details	Click to view details about the table created based on the output lineage.

6.11.35 Open/Close Resource

Functions

You can use the Open/Close Resource node to enable or disable Huawei Cloud services as required.

Parameters

Table 6-157 and Table 6-158 describe the parameters of the Open/Close Resource node.

Table 6-157 Parameters of Open/Close Resource nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Service	Yes	Service to be opened or closed. • **ECS** • **CDM**
Open/Close Resource	Yes	Possible values: • On • Off
Instance	Yes	Object to be opened or closed, for example, to open a CDM cluster.

Table 6-158 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.36 Data Quality Monitor

Functions

The Data Quality Monitor node is used to monitor the quality of running data.

Parameters

Table 6-159 and **Table 6-160** describe the parameters of the Data Quality Monitor node.

Table 6-159 Parameters of Data Quality Monitor nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Job Type	Yes	Data quality job. The following options are available: • Quality job • Comparison job
Quality Job Name	Yes	Name of a quality job created in DataArts Quality. This parameter is mandatory when **Job Type** is **Quality Job**. For details about how to create a quality job, see **Creating Quality Jobs**.
Ignore Quality Job Alarm	Yes	This parameter is mandatory when **Job Type** is **Quality Job**. • **Yes**: If the quality job is in the alarm state, the status of the current node is set to successful and the subsequent nodes continue to be executed. • **No**: If the quality job is in the alarm state, the status of the current node is set to failed.

Parameter	Mandatory	Description
Comparison Job Name	Yes	Name of a comparison job created in DataArts Quality. This parameter is mandatory when **Job Type** is **Comparison Job**. For details about how to create a comparison job, see **Creating a Comparison Job**.
Ignore Comparison Job Alarm	Yes	This parameter is mandatory when **Job Type** is **Comparison Job**. • **Yes**: If the comparison job is in the alarm state, the status of the current node is set to successful and the subsequent nodes continue to be executed. • **No**: If the comparison job is in the alarm state, the status of the current node is set to failed.

Table 6-160 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.

Parameter	Mandatory	Description
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.37 Subjob

Function

The Subjob node is used to call the batch job that does not contain the subjob node.

Parameter

Table 6-161 and **Table 6-162** describe the parameters of the Subjob node.

Table 6-161 Parameters of subjob nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Subjob Name	Yes	Select the name of the subjob to be called. **NOTE** You can only select the name of an existing batch job that does not contain the Subjob node.
Subjob Parameter	Yes/No	• If the subjob parameters are left unspecified, the subjob is executed with its own parameter variables. The **Subjob Parameter Name** of the parent job is not displayed. • If the subjob parameters are specified, the subjob is executed with the configured parameter values. In this case, the **Subjob Parameter Name** of the parent job is displayed, and the data or EL expression configured for the subjob is accessed and replaced according to the environment variable of the parent job.

Table 6-162 Advanced parameters

Parameter	Mandatory	Description
Node Status Polling Interval (s)	Yes	How often the system check completeness of the node. The value ranges from 1 to 60 seconds.
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.38 For Each

Functions

The For Each node specifies a subjob to be executed cyclically and assigns values to variables in a subjob with a dataset.

For details about how to use the For Each node, see **Introduction to the For Each Operator**.

NOTE

When a For Each node is executed once, a specified subjob can be cyclically executed for a maximum of 1,000 times.

If DLI SQL is used as a frontend node, the For Each node supports a maximum of 100 subjobs.

Parameters

Table 6-163 describes the parameters of the For Each node.

Table 6-163 Parameters of the For Each node

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Subjob in a Loop	Yes	Name of the subjob to be executed cyclically.

Parameter	Mandatory	Description
Subjob Parameter Name	No	This parameter is available only when you set job parameters for a cyclic subjob. The parameter name is the variable defined in the subjob. Set the parameter value based on the following rules: • If the cyclic subjob needs to be read and replaced based on the variables of the parent job, set this parameter to an EL expression, for example, **#{Loop.current[0]}** or **#{Loop.current[1]}** which indicates obtaining the first or second value in the current row of the traversed dataset two-dimensional array. For details, see **Loop Embedded Objects**. After a job parameter name is configured for the cyclic subjob, the parameter value can be left empty. • If a cyclic subjob needs to use its own parameter variables, leave this parameter blank. In this case, set values for the parameters of the subjob.
Dataset	Yes	The For Each node needs to define a dataset. The dataset is a two-dimensional array used to cyclically replace variables in a subjob. A row of data in the dataset corresponds to a subjob instance. The dataset may come from the following sources: • Output from upstream nodes, such as the select statements of the Hive SQL, DLI SQL, or Spark SQL node, and echo of the shell node. The EL expression **#{Job.getNodeOutput('preNodeName')}** is used, which means the output of the previous node. • A specified array, for example, two-dimensional array **[['001'],['002'],['003']]** **NOTE** • To transfer **00** and **01** as numbers, set this parameter to **[["00"],["01"]];[[00],[01]];[['00'],['01']]**. • To transfer **00** and **01** as characters, add escape characters, for example, **[["\"00\""],["\"01\""]];[['\'00\''],['\'01\'']]**.
Concurrent Subjobs	Yes	Subjobs generated cyclically can be executed concurrently. You can set the number of concurrent subjobs. **NOTE** If a subjob contains a CDM Job node, set this parameter to 1.

Parameter	Mandatory	Description
Subjob Instance Name Suffix	No	Name of the subjob generated by For Each: For Each node name + underscore (_) + suffix. The suffix is configurable. If the suffix is not configured, the suffix increases in ascending order based on the number.

Table 6-164 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.

Parameter	Mandatory	Description
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.39 SMN

Functions

The SMN node is used to send notifications to users.

Parameters

Table 6-165 and **Table 6-166** describe the parameters of the SMN node.

Table 6-165 Parameters of SMN nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).
Topic Name	Yes	Name of the topic. The topic has been created in SMN.
Message Title	No	Title of the message. The title cannot exceed 512 characters.
Message Type	Yes	Format of the message. ● **Text**: The message is sent in text format. ● **JSON**: The message is sent in JSON format. You can send different messages to types of subscribers. – Manual: You can enter a message in **Message Content**. – Automatic: Click **Generate JSON Message**. In the displayed dialog box, enter a message and select a protocol. ● **Template**: The message is sent in template format, that is, in fixed format. The variables can be processed by tags. – Manual: You can enter a message in **Message Content**. – Automatic: Click **Generate Template Message**. In the displayed dialog box, select a template name and set the value of **tag**.

Parameter	Mandatory	Description

Message Content	Yes	Message content to be provided. The requirements for entering different types of messages are as follows:
		● **Text**: The size cannot exceed 10 KB.
		● **JSON**: The JSON message must contain the Default protocol and the size cannot exceed 10 KB. Example:
		{
		"default": "Dear Sir or Madam, this is a default message.",
		"email": "Dear Sir or Madam, this is an email message.",
		"http": "{'message':'Dear Sir or Madam, this is an HTTP message.'}",
		"https": "{'message':'Dear Sir or Madam, this is an HTTPS message.'}",
		"sms": "This is an SMS message."
		}
		● **Template**: The size cannot exceed 10 KB. Example:
		"message_template_name":"confirm_message",
		"tags":{
		"topic_urn":"urn:smn:regionId:xxxx:SMN_01"
		}
		In the preceding information, **message_template_name** indicates the template name, and **tags** indicates all tags in the template.
		For details about how to configure SMN, see section the *Simple Message Notification User Guide*.

Table 6-166 Advanced parameters

Parameter	Mandatory	Description
Max. Node Execution Duration	Yes	Execution timeout interval for the node. If retry is configured and the execution is not complete within the timeout interval, the node will be executed again.

Parameter	Mandatory	Description
Retry upon Failure	Yes	Whether to re-execute a node if it fails to be executed. Possible values: • **Yes**: The node will be re-executed, and the following parameters must be configured: – **Retry upon Timeout** – **Maximum Retries** – **Retry Interval (seconds)** • **No**: The node will not be re-executed. This is the default setting. **NOTE** If retry is configured for a job node and the timeout duration is configured, the system allows you to retry a node when the node execution times out. If a node is not re-executed when it fails upon timeout, you can go to the **Default Configuration** page to modify this policy. **Retry upon Timeout** is displayed only when **Retry upon Failure** is set to **Yes**.
Policy for Handling Subsequent Nodes If the Current Node Fails	Yes	Operation that will be performed if the node fails to be executed. Possible values: • **Suspend execution plans of the subsequent nodes**: stops running subsequent nodes. The job instance status is **Failed**. • **End the current job execution plan**: stops running the current job. The job instance status is **Failed**. • **Go to the next node**: ignores the execution failure of the current node. The job instance status is **Failure ignored**. • **Suspend the current job execution plan**: If the current job instance is in abnormal state, the subsequent nodes of this node and the subsequent job instances that depend on the current job are in waiting state.
Enable Dry Run	No	If you select this option, the node will not be executed, and a success message will be returned.

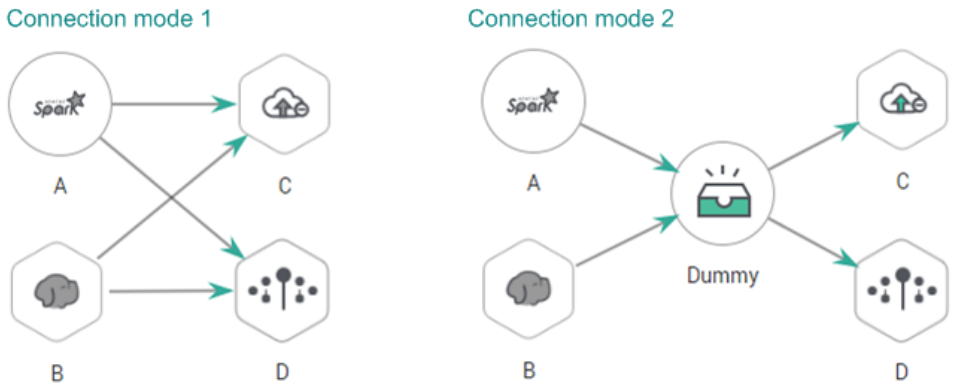
Parameter	Mandatory	Description
Task Groups	No	Select a task group. If you select a task group, you can control the maximum number of concurrent nodes in the task group in a fine-grained manner in scenarios where a job contains multiple nodes, a data patching task is ongoing, or a job is rerunning.

6.11.40 Dummy

Functions

The Dummy node is empty and does not perform any operations. It is used to simplify the complex connection relationships of nodes. **Figure 6-134** shows an example.

Figure 6-134 Connection modes



Parameters

Table 6-167 describes the parameter of Dummy nodes.

Table 6-167 Parameter of Dummy nodes

Parameter	Mandatory	Description
Node Name	Yes	Name of a node. The name must contain 1 to 128 characters, including only letters, numbers, underscores (_), hyphens (-), slashes (/), less-than signs (<), and greater-than signs (>).

6.12 EL Expression Reference

6.12.1 Expression Overview

Node parameter values in a DataArts Factory job can be dynamically generated based on the running environment by using Expression Language (EL). You can determine whether to execute this node based on the input parameters of the pipeline and the output of the upstream node. EL uses simple arithmetic and logic to calculate and references embedded objects, including job objects and tool objects.

Job object: provides properties and methods of obtaining the output message, job scheduling plan time, and job execution time of the previous node in a job.

Tool job: Provides methods of operating character strings, time, and JSON. For example, truncating a substring from a string or formatting time.

Syntax

Expression syntax:

```
#{expr}
```

In the preceding information, **expr** indicates an expression. **#** and **{}** are common operators used in EL, allowing you to access job properties using embedded objects.

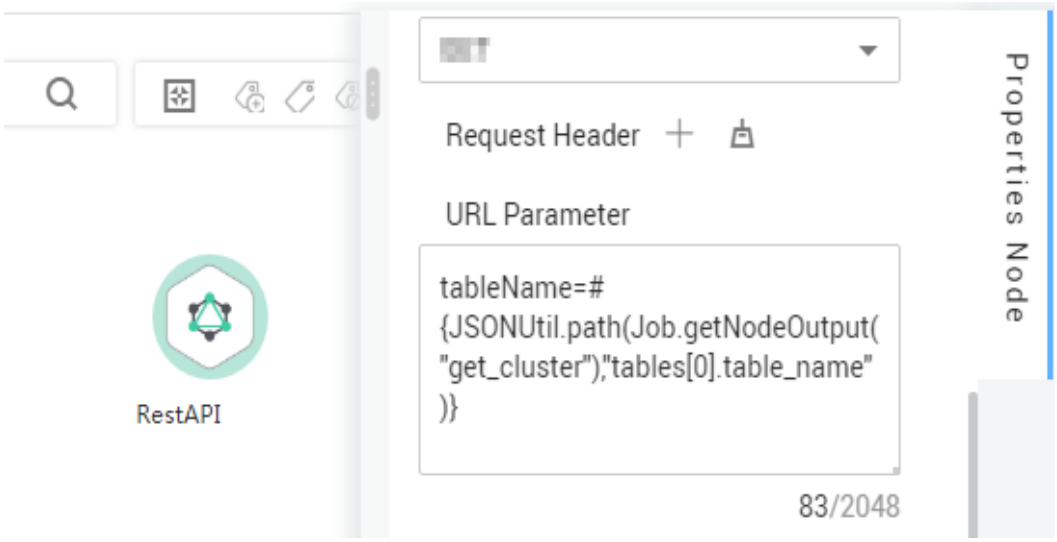
Example

In the **URL** parameter of the Rest Client node, use expression **tableName=#{JSONUtil.path(Job.getNodeOutput("get_cluster"),"tables[0].table_name")}**, as shown in **Figure 6-135**.

Expression description:

1. **Job.getNodeOutput("get_cluster")** is used to obtain the execution result of the **get_cluster** node in the job. The execution result is a JSON character string.
2. **tables[0].table_name** is used to obtain the value of a field in the JSON character string.

Figure 6-135 Expression example



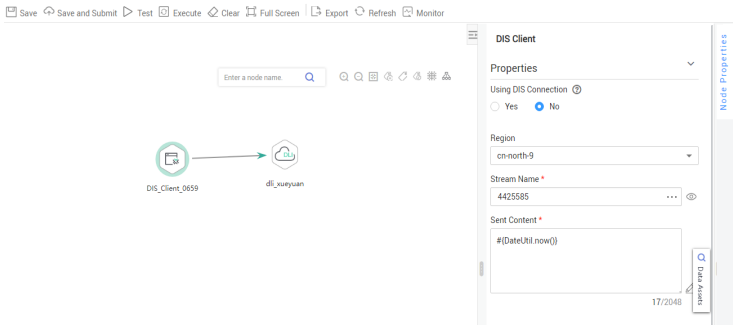
EL expressions are widely used in data development. For details, see **Best Practices**.

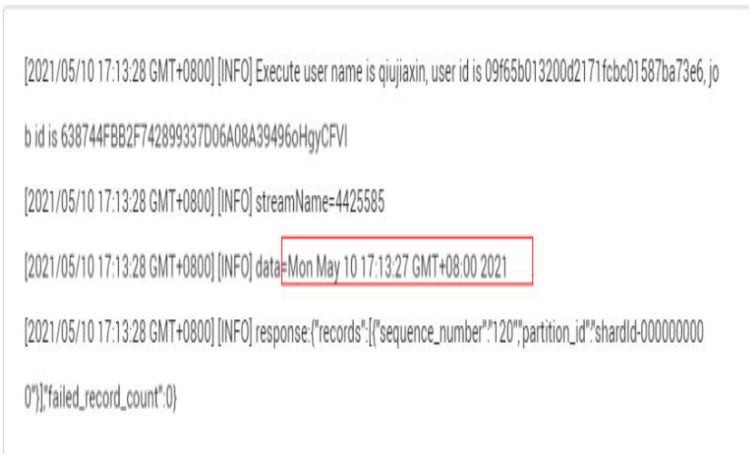
Debugging Methods

You can debug EL expressions using the following methods.

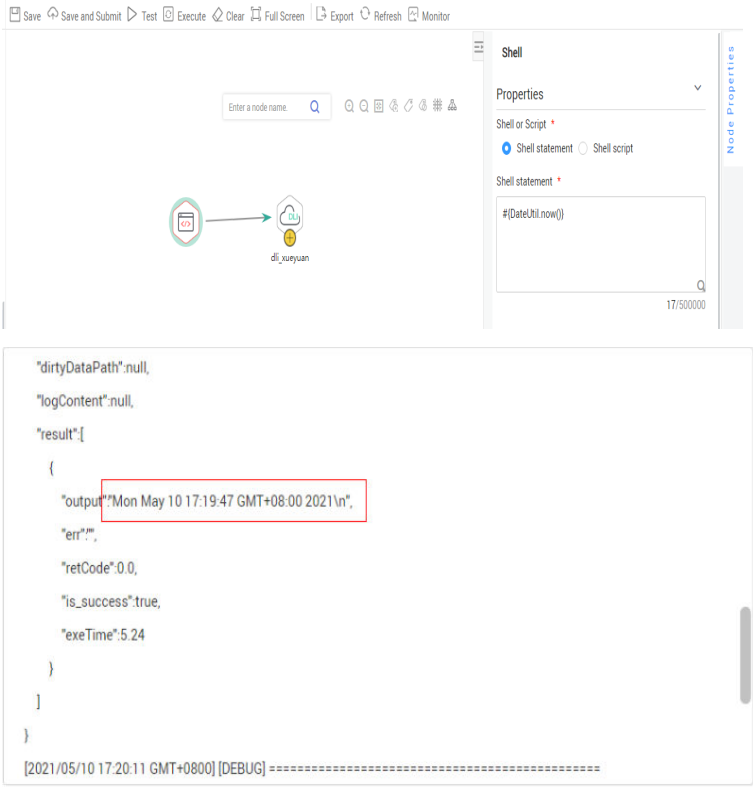
This section uses the #{DateUtil.now()} expression as an example.

1. Use the DIS Client node.
   - Prerequisites: A DIS stream is available.
   - Method: Select the DIS Client node, write the EL expression in the data to be sent, and click **Test**. Then right-click the node to view the log. The value of the EL expression is printed in the log.

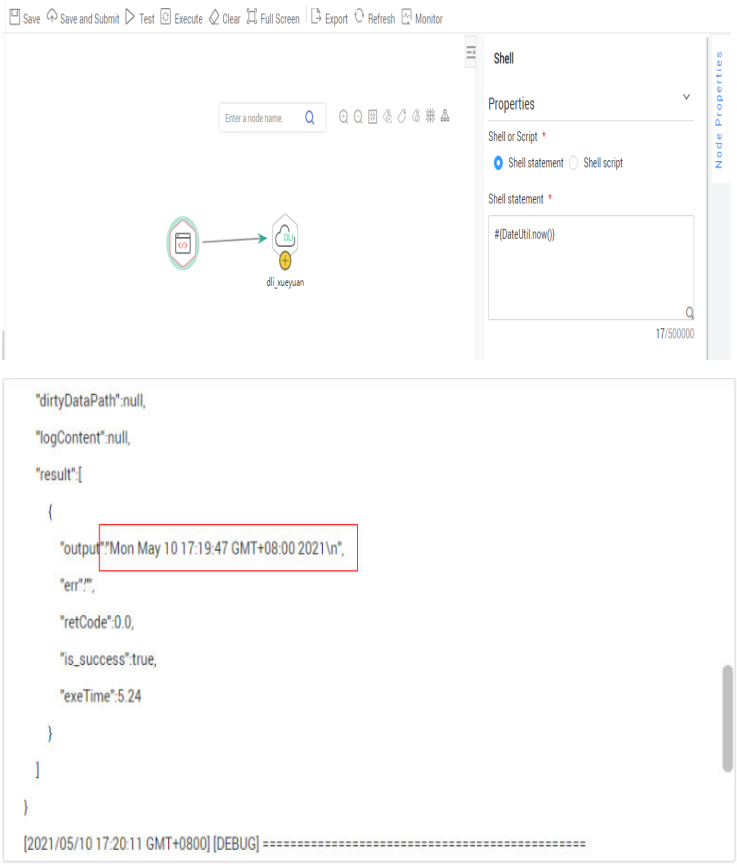




2. Use the Kafka Client node.
   - Prerequisites: An MRS cluster with the Kafka component is available.
   - Method: Select the Kafka Client node, write the EL expression in the data to be sent, and click **Test**. Then right-click the node to view the log. The value of the EL expression is printed in the log.

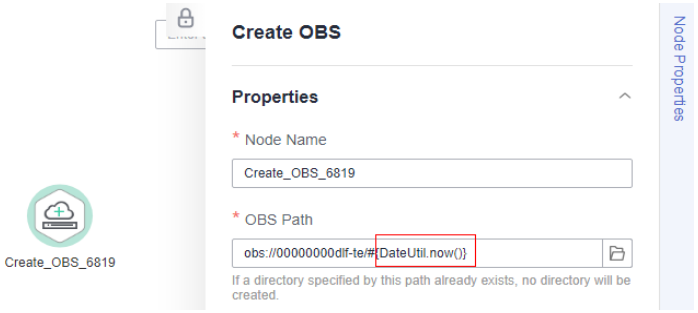


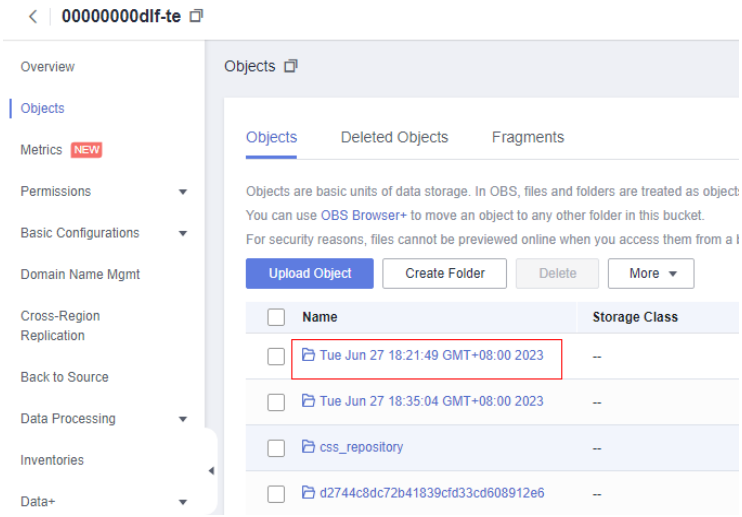
3. Use the shell node.
   - Prerequisites: An ECS is available.
   - Method: Create a host connection, print the EL expression using echo, and click **Test**. Then view the log. The value of the EL expression is printed in the log.



4. Use the Create OBS node.

   If none of the preceding methods is available, use the Create OBS node and create an OBS path with the value of the EL expression as its name. You can click **Test** and go to the OBS console to view the name of the created path.





6.12.2 Basic Operators

EL supports most of the arithmetic and logic operators provided by Java.

Operator List

Table 6-168 Basic operators

Operator	Description
.	Accesses a Bean property or a mapping entry.
[]	Accesses an array or linked list.
()	Organizes a subexpression to change priority.
+	Plus sign
-	Minus or negative sign
*	Multiplication sign
/ or div	Division sign
% or mod	Modulo
== or eq	Test whether equal to.
!= or ne	Test whether unequal to.
< or lt	Test whether less than.
> or gt	Test whether greater than.
<= or le	Check whether less than or equal to.
>= or ge	Test whether greater than or equal to.
&& or and	Test logic and.

Operator	Description
|| or or	Test logic or.
! or not	Test negation.
empty	Test whether empty.
?:	The expression is similar to if else. If the statement in front of ? is true, the value of the expression between ? and : is returned. Otherwise, the value following : is returned.

Example

If variable a is empty, default is returned. If variable a is not empty, a itself is returned. The EL expression is as follows:

```
#{empty a?"default":a}
```

6.12.3 Date and Time Mode

The date and time in the EL expression can be displayed in a user-specified format. The date and time format is specified by the date and time mode character string. The date and time mode character string consists of letters from A to Z and from a to z, as shown in **Table 6-169**.

Table 6-169 Letter description

Letter	Description	Example
G	Epoch	AD
y	Year	2001
M	Month in a year	July or 07
d	Day in a month	10
h	Hour in the 12-hour clock (1 to 12)	12
H	Hour in the 24-hour clock (0 to 23)	22
m	Minute	30
s	Second	55
S	Millisecond	234
E	Day of a week	Mon, Tue, Wed, Thu, Fri, Sat, or Sun
D	Date in the year	360

Letter	Description	Example
F	Day in a week of a month	2(second Wed. in July)
w	Week in a year	40
W	Week in a month	1
a	A.M. /P.M.	PM
k	Hour in the 24-hour clock (1 to 24)	24
K	Hour in the 12-hour clock (0 to 11)	10
z	Time zone	Eastern Standard Time
'	Text delimiter	None
''	Single quotation mark	No example

NOTE

The date and time mode is generally used in DateUtil embedded objects and Job embedded objects. For more examples of how the date and time mode is used, see **DateUtil Embedded Objects** and **Job Embedded Objects**.

Example

To obtain the date of the day before the planned scheduling time of a job, use the following EL expression:

```
#{DateUtil.format(DateUtil.addDays(Job.planTime,-1),"yyyy-MM-dd")}
```

6.12.4 Env Embedded Objects

An Env embedded object provides a method of obtaining an environment variable value.

Method

Table 6-170 Method description

Method	Description	Example
String get(String name)	Obtains the value of a specified environment variable.	To obtain the value of the environment variable **test**, run the following command: #{Env.get("test")}

Example

The EL expression used to obtain the value of environment variable **test** is as follows:

```
#{Env.get("test")}
```

6.12.5 Job Embedded Objects

A job object provides properties and methods of obtaining the output message, job scheduling plan time, and job execution time of the previous node in a job.

Properties and Methods

Table 6-171 Property description

Property	Type	Description
name	String	Job name.
planTime	java.util.Date	Job scheduling plane time, that is, the time configured for periodic scheduling, for example, to schedule a job at 1:01 a.m. every day.
startTime	java.util.Date	Job execution time. It may be the same as or later than the planTime (because the job engine is busy).
eventData	String	Message obtained from the stream when the event-driven scheduling is used.
projectId	String	ID of the project where the DataArts Factory module is located.

Table 6-172 Method description

Method	Description	Example
String getNodeStatus(String nodeName)	Obtains the running status of a specified node. If the node runs properly, success is returned. If the node fails to run, fail is returned. For example, to check whether a node is running successfully, you can use the following command, where **test** indicates the node name: **#{(Job.getNodeStatus("test")) == "success" }**	Obtain the running status of the test node: #{Job.getNodeStatus("test")}

Method	Description	Example
String getNodeOutput(String nodeName)	Obtains the output of a specified node. This method can only obtain the output of the previous dependent node.	• Obtain the output of the test node: #{Job.getNodeOutput("test")} • If the previous node has no execution result, the output is null. • If the output of a node is a field, the output result is in the format like **[["000"]]**. In this case, you can use the EL expression to split the string result and obtain the field value output by the previous node. Note that the output result type is string. If you want to output the original data type, you need to use the EL expression of the For Each node and the loop embedded objects supported by the node. #{StringUtil.split(StringUtil.split(StringUtil.split(Job.getNodeOutput("Name of the previous node"),"]")[0],"[")[0],"\\"")[0]} • If the output of a node contains two or more fields, the output result is in the format like **[["000"], ["001"]]**. In this case, you need to obtain the output result using the EL expression of the For Each node and the loop embedded objects supported by the node, for example, **#{Loop.current[0]}**.

Method	Description	Example
String getParam(String key)	Obtains job parameters. This method only obtains the parameter values configured for the current job, but not parameter values passed from the parent job or the global variables configured for the workspace. To obtain the parameter values passed from the parent job and the global variables configured for the workspace, you are advised to use the ${job_param_name} expression.	Obtain the value of the **test** parameter: #{Job.getParam("test")}
String getPlanTime(String pattern)	Obtains the plan time character string in a specified pattern. Pattern indicates the date and time mode. For details, see **Date and Time Mode**.	Obtain the planned job scheduling time, which is accurate to millisecond: #{Job.getPlanTime("yyyy-MM-dd HH:mm:ss:SSS")}
String getYesterday(String pattern)	Obtains the time character string of the day before the plan time. Pattern indicates the date and time mode. For details, see **Date and Time Mode**.	Obtain the time on the previous day of the planned job scheduling time, which is accurate to date: #{Job.getYesterday("yyyy-MM-dd HH:mm:ss:SSS")}
String getLastHour(String pattern)	Obtains the time character string of last hour before the plan time. Pattern indicates the date and time mode. For details, see **Date and Time Mode**.	Obtain the time one hour before the planned job scheduling time, which is accurate to hour: #{Job.getLastHour("yyyy-MM-dd HH:mm:ss:SSS")}

Method	Description	Example
String getRunningData(String nodeName)	Obtains the data recorded during the running of a specified node. Currently, only the IDs of the jobs running using SQL statements on the DLI SQL node can be obtained. This method can only obtain the output of the previous dependent node. For example, to obtain the job ID of the third statement on DLI node **DLI_INSERT_DATA**, run the following command: **#{JSONUtil.path(Job.getRunningData("DLI_INSERT_DATA"),"jobIds[2]")}**.	Obtain the ID of the job run by the third statement in the test of the DLI SQL node: #{JSONUtil.path(Job.getRunningData("test"),"jobIds[2]")}
String getInsertJobId(String nodeName)	Returns the job ID in the first Insert SQL statement of the specified DLI SQL or Transform Load node. If the **nodeName** parameter is not specified, the job ID in the first DLI Insert SQL statement of the DLI SQL node is obtained. If the job ID cannot be obtained, the **null** value is returned.	Obtain the ID of job run by the first Insert SQL statement in the test of the DLI SQL node: #{Job.getInsertJobId("test")}
String getPreviousWorkday(Integer num, String pattern)	Returns the time string of the **num** working day before a planned time specified by **pattern**. The value of **num** must be a positive integer. If no result that meets the specified condition is found, null is returned. This EL expression is suitable for selecting custom dates in a calendar to schedule jobs.	Obtains the date of the fifth working day before a specified day. #{Job.getPreviousWorkday(5, "yyyyMMdd")}

Method	Description	Example
String getPreviousNonWorkday(Integer num, String pattern)	Returns the time string of the **num** non-working day before a planned time specified by **pattern**. The value of **num** must be a positive integer. If no result that meets the specified condition is found, null is returned. This EL expression is suitable for selecting custom dates in a calendar to schedule jobs.	Obtains the date of the first non-working day before a specified day. #{Job.getPreviousNonWorkday(1, "yyyyMMdd")}

Example 1

The expression used to obtain the output of node **test** in the job is as follows:

```
#{Job.getNodeOutput("test")}
```

6.12.6 StringUtil Embedded Objects

A StringUtil embedded object provides methods of operating character strings, for example, truncating a substring from a character string.

StringUtil is implemented through org.apache.commons.lang3.StringUtils. For details about how to use the object, see the **Apache Commons documentation**.

Example 1

If variable a is character string No.0010, the substring after **.** is returned. The EL expression is as follows:

```
#{StringUtil.substringAfter(a,".")}
```

Example 2

If variable b is string No,0020, the substring after **,** is returned. The EL expression is as follows:

```
#{StringUtil.split(b,',')[1]}
```

Example 3

If the output of a node is a field, the output result is shown in [["000"]]. The second node references the output of the first node. In this case, the EL expression can be used to split the string result and obtain the field value output by the previous node.

```
#{StringUtil.split(StringUtil.split(StringUtil.split(Job.getNodeOutput("Name of the previous node"),"]")
[0],"[")[0],"\\"")[0]}
```


Example 4

If the output of the previous SQL node is [["11"]], the following EL expression can be used to obtain value "11":

#{StringUtil.getDigits(Job.getNodeOutput("nodeName"))}

Example 5

Returns the digits extracted from a string.

String getDigits(String str)

For example, if str is "1123~45", "112345" is returned; if str is "abc", "" is returned; if str is "12345", "12345" is returned.

6.12.7 DateUtil Embedded Objects

A DateUtil embedded object provides methods of formatting time and calculating time.

Methods

Table 6-173 Method description

Method	Description	Example
String format(Date date, String pattern)	Formats Date to character strings according to the specified pattern.	Convert the planned job scheduling time to the millisecond format. #{DateUtil.format(Job.planTime,"yyyy-MM-dd HH:mm:ss:SSS")} Subtracts one day from the planned job scheduling time and convert the time to the week format. • #{DateUtil.format(DateUtil.addDays(Job.planTime,-1),"yyyyw")} If **Job.planTime** is January 7, 2024, value **20241** is returned. • #{DateUtil.format(DateUtil.addDays(Job.planTime,-1),"yyyyww")} If **Job.planTime** is January 7, 2024, value **202401** is returned.

Method	Description	Example
Date addMonths(Date date, int amount)	After the specified number of months is added to Date, the new Date object is returned. The amount can be a negative number.	Subtract one month from the planned job scheduling time and convert the time to the month format. #{DateUtil.format(DateUtil.addMonths(Job.planTime,-1),"yyyy-MM")}
Date addDays(Date date, int amount)	After the specified number of days is added to Date, the new Date object is returned. The amount can be a negative number.	Subtracts one day from the planned job scheduling time and convert the time to the yyyy-MM-dd format. #{DateUtil.format(DateUtil.addDays(Job.planTime,-1),"yyyy-MM-dd")} Subtracts one day from the planned job scheduling time and convert the time to the week format. • #{DateUtil.format(DateUtil.addDays(Job.planTime,-1),"yyyyw")} If **Job.planTime** is January 7, 2024, value **20241** is returned. • #{DateUtil.format(DateUtil.addDays(Job.planTime,-1),"yyyyww")} If **Job.planTime** is January 7, 2024, value **202401** is returned.
Date addHours(Date date, int amount)	After the specified number of hours is added to Date, the new Date object is returned. The amount can be a negative number.	Subtract one hour from the planned job scheduling time and convert the time to the hour format. #{DateUtil.format(DateUtil.addHours(Job.planTime,-1),"yyyy-MM-dd HH")}
Date addMinutes(Date date, int amount)	After the specified number of minutes is added to Date, the new Date object is returned. The amount can be a negative number.	Subtract one minute from the planned job scheduling time and convert the time to the minute format. #{DateUtil.format(DateUtil.addMinutes(Job.planTime,-1),"yyyy-MM-dd HH:mm")}

Method	Description	Example
int getDay(Date date)	Obtains the day from the date. For example, if the date is 2018-09-14, 14 is returned.	Obtain the day from the job scheduling plan. #{DateUtil.getDay(Job.planTime)}
int getMonth(Date date)	Obtains the month from the date. For example, if the date is 2018-09-14, 9 is returned.	Obtain the month from the date. #{DateUtil.getMonth(Job.planTime)}
int getQuarter(Date date)	Obtains the quarter from the date. For example, if the date is 2018-09-14, 3 is returned.	Obtain the quarter from the date. #{DateUtil.getQuarter(Job.planTime)}
int getYear(Date date)	Obtains the year from the date. For example, if the date is 2018-09-14, 2018 is returned.	Obtain the year from the date. #{DateUtil.getYear(Job.planTime)}
Date now()	Returns the current time.	Return the current time accurate to second. #{DateUtil.format(DateUtil.now(),"yyyy-MM-dd HH:mm:ss")}
long getTime(Date date)	Converts a time of the date type to one of the long type.	Convert the planned job scheduling time to a timestamp. #{DateUtil.getTime(Job.planTime)}
Date parseDate(String str, String pattern)	Converts the character string to the date by pattern. The pattern is the date and time mode. For details, see **Date and Time Mode**.	Convert the job start time string to a time accurate to second. #{DateUtil.parseDate(Job.getPlanTime("yyyy-MM-dd HH:mm:ss:SSS"),"yyyy-MM-dd HH:mm:ss")}

Example

The previous day of the job scheduling plan time is used as the subdirectory name to generate an OBS path. The EL expression is as follows:

```
#{"obs://test/"+DateUtil.format(DateUtil.addDays(Job.planTime,-1),"yyyy-MM-dd")}
```

6.12.8 JSONUtil Embedded Objects

A JSONUtil embedded object provides JSON object methods.

Methods

Table 6-174 Method description

Method	Description	Example
Object parse(String jsonStr)	Converts a JSON character string into an object.	Assume that variable a is a JSON string. Use the following EL expression to convert the JSON string into an object: #{JSONUtil.parse(a)}
String toString(Object jsonObject)	Converts an object to a JSON character string.	Assume that variable b is an object. Use the following EL expression to convert the object into a JSON string: #{JSONUtil.toString(b)}

Object path(String jsonStr,String jsonPath)	Returns the field value in a path specified by the JSON character string. This method is similar to XPath and can be used to retrieve or set JSON by path. You can use . or [] in the path to access members and values. For example, tables[0].table_name.	The content of variable str is as follows:
		{
		"cities": [{
		"name": "city1",
		"areaCode": "1000"
		},
		{
		"name": "city2",
		"areaCode": "2000"
		},
		{
		"name": "city3",
		"areaCode": "3000"
		}]
		}
		The expression for obtaining the area code of city1 is as follows:
		#{JSONUtil.path(str,"cities[0].areaCode")}

Example

The content of variable str is as follows:

```
{
	"cities": [{
		"name": "city1",
		"areaCode": "1000"
	},
	{
		"name": "city2",
		"areaCode": "2000"
	},
```



```
        {
                "name": "city3",
                "areaCode": "3000"
        }]
}
```

The expression for obtaining the area code of city1 is as follows:

```
#{JSONUtil.path(str,"cities[0].areaCode")}
```

6.12.9 Loop Embedded Objects

You can use Loop embedded objects to obtain data from the For Each node.

Property

Table 6-175 Property description

Property	Type	Description	Example
dataArray	String	**Loop.dataArray** indicates the two-dimensional array defined in the dataset of the For Each node. Generally, the format is **#{Loop.dataArray[0][0]}** or **#{Loop.dataArray[0][1]}**. **[0][0]** indicates the first value in the first row of the array, and **[0][1]** indicates the second value in the first row, and so on.	The value of **Subjob Parameter** for the For Each node indicates that the first value in the second row of the two-dimensional array in the dataset is always used in the For Each loop. #{Loop.dataArray[1][0]}
current	String	For Each nodes process data in a dataset by row. **Loop.current** indicates a row of a two-dimensional array defined in the dataset of the For Each node. This row is a one-dimensional array. Generally, the format is similar to **#{Loop.current[0]}**, **#{Loop.current[1]}**, or others. **[0]** indicates the first value in the current row, **[1]** indicates the second value in the current row, and so on.	The value of **Subjob Parameter** for the For Each node indicates that the second value in the traversed row of the two-dimensional array in the dataset is always used in the loop traversal of the For Each node. #{Loop.current[1]}

Property	Type	Description	Example
offset	Int	Current offset of the For Each node, starting from 0. Loop.dataArray[Loop.offset] = Loop.current.	Obtain the current offset of the For Each loop, that is, the number of traversals, starting from 0. #{Loop.offset}

Example

To obtain the second value of a row that is being processed, use the following EL expression:

```
#{Loop.current[1]}
```

6.12.10 OBSUtil Embedded Objects

The OBSUtil embedded objects provide a series of OBS operation methods, for example, checking whether an OBS file or directory exists.

Methods

Table 6-176 Method description

Method	Description	Example
boolean isExistOBSPath(String obsPath)	Check whether the OBS file or the OBS directory that ends with a slash (/) exists. If the file or directory exists, **true** is returned. If not, **false** is returned.	• The following is the EL expression for checking whether the OBS directory that ends with a slash (/) exists: #{OBSUtil.isExistOBSPath("obs://test/jobs/")} • The following is the EL expression for checking whether the OBS file exists: #{OBSUtil.isExistOBSPath("obs://test/jobs/job.log")}

Examples

- The following is the EL expression for checking whether the OBS directory that ends with a slash (/) exists:
  #{OBSUtil.isExistOBSPath("obs://test/jobs/")}
- The following is the EL expression for checking whether the OBS file exists:
  #{OBSUtil.isExistOBSPath("obs://test/jobs/job.log")}

6.12.11 Examples of Common EL Expressions

This section describes common EL expressions and examples.

Table 6-177 Common EL expressions

Method	Description	Example
String getNodeStatus(String nodeName)	Obtains the running status of a specified node. If the node runs properly, success is returned. If the node fails to run, fail is returned. For example, to check whether a node is running successfully, you can use the following command, where **test** indicates the node name: **#{(Job.getNodeStatus("test")) == "success" }**	Obtain the running status of the test node: #{Job.getNodeStatus("test")}

Method	Description	Example
String getNodeOutput(String nodeName)	Obtains the output of a specified node. This method can only obtain the output of the previous dependent node.	• Obtain the output of the test node: #{Job.getNodeOutput("test")} • If the previous node has no execution result, the output is null. • If the output of a node is a field, the output result is in the format like **[["000"]]**. In this case, you can use the EL expression to split the string result and obtain the field value output by the previous node. Note that the output result type is string. If you want to output the original data type, you need to use the EL expression of the For Each node and the loop embedded objects supported by the node. #{StringUtil.split(StringUtil.split(StringUtil.split(Job.getNodeOutput("Name of the previous node"),"]")[0],"[")[0],"\\"")[0]} • If the output of a node contains two or more fields, the output result is in the format like **[["000"], ["001"]]**. In this case, you need to obtain the output result using the EL expression of the For Each node and the loop embedded objects supported by the node, for example, **#{Loop.current[0]}**.

Method	Description	Example
String getParam(String key)	Obtains job parameters. This method only obtains the parameter values configured for the current job, but not parameter values passed from the parent job or the global variables configured for the workspace. To obtain the parameter values passed from the parent job and the global variables configured for the workspace, you are advised to use the ${job_param_name} expression.	Obtain the value of the **test** parameter: #{Job.getParam("test")}
String getPlanTime(String pattern)	Obtains the plan time character string in a specified pattern. Pattern indicates the date and time mode. For details, see **Date and Time Mode**.	Obtain the planned job scheduling time, which is accurate to millisecond: #{Job.getPlanTime("yyyy-MM-dd HH:mm:ss:SSS")}
String getYesterday(String pattern)	Obtains the time character string of the day before the plan time. Pattern indicates the date and time mode. For details, see **Date and Time Mode**.	Obtain the time on the previous day of the planned job scheduling time, which is accurate to date: #{Job.getYesterday("yyyy-MM-dd HH:mm:ss:SSS")}
String getLastHour(String pattern)	Obtains the time character string of last hour before the plan time. Pattern indicates the date and time mode. For details, see **Date and Time Mode**.	Obtain the time one hour before the planned job scheduling time, which is accurate to hour: #{Job.getLastHour("yyyy-MM-dd HH:mm:ss:SSS")}
Date addDays(Date date, int amount)	After the specified number of days is added to Date, the new Date object is returned. The amount can be a negative number.	Subtracts one day from the planned job scheduling time and convert the time to the yyyy-MM-dd format. #{DateUtil.format(DateUtil.addDays(Job.planTime,-1),"yyyy-MM-dd")}
int getDay(Date date)	Obtains the day from the date. For example, if the date is 2018-09-14, 14 is returned.	Obtain the day from the job scheduling plan. #{DateUtil.getDay(Job.planTime)}

Method	Description	Example
Date now()	Returns the current time.	Return the current time accurate to second. #{DateUtil.format(DateUtil.now(),"yyyy-MM-dd HH:mm:ss")}

Object path(String jsonStr,String jsonPath)	Returns the field value in a path specified by the JSON character string. This method is similar to XPath and can be used to retrieve or set JSON by path. You can use . or [] in the path to access members and values. For example, tables[0].table_name.	The content of variable str is as follows:
		{
		"cities": [{
		"name": "city1",
		"areaCode": "1000"
		},
		{
		"name": "city2",
		"areaCode": "2000"
		},
		{
		"name": "city3",
		"areaCode": "3000"
		}]
		}
		The expression for obtaining the area code of city1 is as follows:
		#{JSONUtil.path(str,"cities[0].areaCode")}

current	For Each nodes process data in a dataset by row. **Loop.current** indicates a row of a two-dimensional array defined in the dataset of the For Each node. This row is a one-dimensional array. Generally, the format is similar to **#{Loop.current[0]}**, **#{Loop.current[1]}**, or others. **[0]** indicates the first value in the current row, **[1]** indicates the second value in the current row, and so on.	The value of **Subjob Parameter** for the For Each node indicates that the second value in the traversed row of the two-dimensional array in the dataset is always used in the loop traversal of the For Each node. #{Loop.current[1]}

6.12.12 EL Expression Use Examples

With this example, you can understand how to use EL expressions in the following applications:

- Using variables in the SQL script of DataArts Factory
- Transferring parameters to SQL script variables?
- Using EL expressions in parameters?

Context

Use the job orchestration and job scheduling functions to generate daily transaction statistics reports according to transaction details tables.

The tables involved in this example are as follows:

- trade_log: This table records data generated in each transaction.
- trade_report: This table is generated based on trade_log and records the daily transaction summary.

Prerequisites

- A DLI data connection named **dli_demo** has been created.

  If this data connection is not created, create one. For details, see **Managing Data Connections**.
- A database named **dli_db** has been created in DLI.

  If this database is not created, create one. For details, see **Creating a Database**.
- Tables **trade_log** and **trade_report** have been created in the **dli_db** database.

  If the tables are not created, create them. For details, see **Creating a Table**.

Procedure

Step 1 Create and develop a SQL script.

1. In the left navigation pane of the DataArts Factory console, choose **Data Development** > **Develop Script** and click **+ DLI**.
2. Go to the SQL script development page and set the data connection, database, and resource queue on the script property bar.

   Figure 6-136 Property bar

3. Enter the following SQL statements in the script editor:

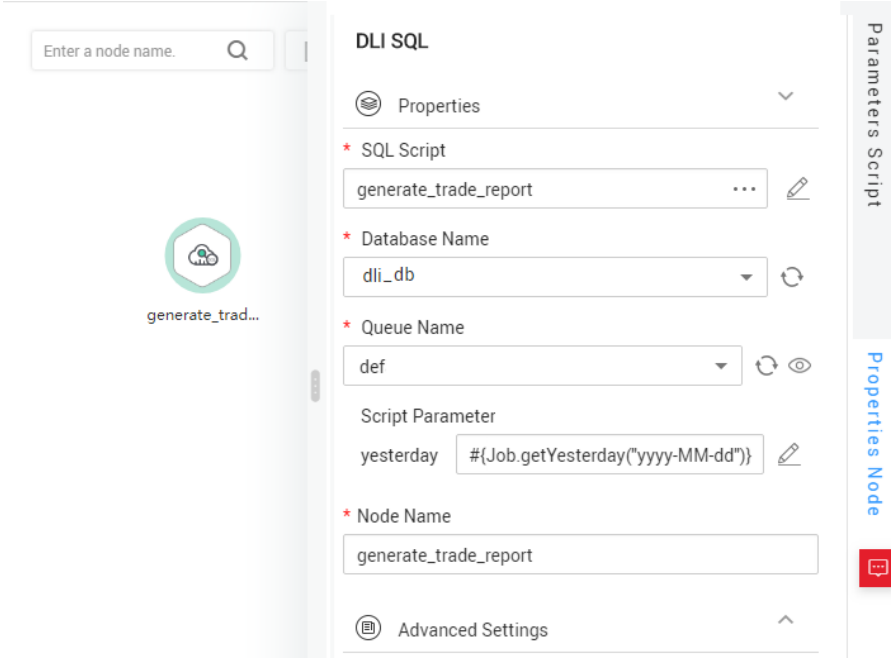
   ```
   INSERT OVERWRITE TABLE trade_report
   SELECT
    sum(trade_count),
    '${yesterday}'
   FROM
    trade_log
   where
    date_format(trade_time, 'yyyy-MM-dd') = '${yesterday}'
   ```

4. Click  and set the script name to **generate_trade_report**.

Step 2 Create and develop a job.

1. In the left navigation pane on the DataArts Factory console, choose **Data Development** > **Develop Job** and click **Create Job** to create an empty job named **job**.
2. Go to the job development page, drag the DLI SQL node to the canvas, click the icon, and configure node properties.

Figure 6-137 Node properties



Description of key properties:

- SQL Script: SQL script **generate_trade_report** that is developed in **Step 1**.
- Database Name: Database configured in SQL script **generate_trade_report**.
- Queue Name: Resource queue configured in SQL script **generate_trade_report**.
- Script Parameter: Parameter **yesterday** configured in SQL script **generate_trade_report**. Enter the following EL expression as the parameter values:

  ```
  #{Job.getYesterday("yyyy-MM-dd")}
  ```

  Expression Description: The job object uses the getYesterday method to obtain the time of the day before the job plan execution time. The time format is yyyy-MM-dd.

  If the job plan time is 2018/9/26 01:00:00, the calculation result of this expression is 2018-09-25. The calculation result will replace the value of parameter ${yesterday} in the SQL script. The SQL statements after the replacement are as follows:

  ```
  INSERT OVERWRITE TABLE trade_report
  SELECT
    sum(trade_count),
    '2018-09-25'
  FROM
    trade_log
  where
    date_format(trade_time, 'yyyy-MM-dd') = '2018-09-25'
  ```

3. Click  to test the running job.
4. After the job test is complete, click  to save the job configuration.

----End

More Examples

EL expressions are widely used in data development. For details, see **Best Practices**.

6.13 Simple Variable Set

The simple variable set provides a series of customized variables. Customized parameters are automatically replaced with specific values based on the service date, plan time, and parameter value format of task scheduling. In this way, parameters can be dynamically replaced during task scheduling.

Currently, the system supports the customization of three types of parameters: service date, plan time, and service ID.

- The business date refers to the day before the expected scheduling running time of a task within the scheduling time, that is, yesterday. The business date is accurate to day. You can obtain the service date from ${yyyymmdd}. Generally, the service date is the date when the plan time is located minus 1.
- The plan time refers to the time point (that is, the current day) when a task is expected to be scheduled within the scheduling time. The plan time is accurate to seconds. The planned time can be obtained through $[yyyymmddhh24miss].
- The service ID parameter includes the job ID and the instance ID generated by the job, which can be obtained through $job_id and $instance_id.

NOTICE

To use simple variable sets, you must enable this function by referring to **Configuring a Default Item > Use Simple Variable Set**.

Service Date Parameter

The service date refers to the day before the expected scheduling running time of a task within the scheduling time, that is, yesterday. For example, if the scheduling date is January 1, 2023, the service date is December 31, 2022. This parameter is a time parameter generated based on the combination of yyyy, yy, mm, and dd. The format of this parameter can be customized. for example, ${yyyy}, ${yyyymm}, ${yyyymmdd}, and ${yyyy-mm-dd}.

- yyyy: indicates a 4-digit year. The value is the year of the service date.
- yy: indicates a 2-digit year. The value is the last two digits of the year of the service date.
- mm: indicates the month. The value is the month of the service date.
- dd: indicates the day. The value is the day of the service date.

For details about how to obtain the time data N years ago, N months ago, and N days ago, see **Table 6-178**. The parameter can only be accurate to year, month, and day. The hour, minute, and second formats are not supported.

Table 6-178 Parameters for obtaining the service date

Business Date Scenario	Method
Previous/Next N Years	${yyyy±N}
Previous/Next N Months	${yyyymm±N}
N weeks before/after	${yyyymmdd±7*N}
N days before/after	${yyyymmdd±N}
N years before/after (yy format)	${yy±N}

Plan Time Parameters

The planned time refers to the time when a task is expected to be scheduled and run within the scheduling time (that is, the current day). This parameter is a time parameter generated based on the combination of yyyy, yy, mm, dd, hh24, mi, and ss. The format of this parameter can be customized. For example, $[yyyymmdd], $[yyyy-mm-dd], $[hh24miss], $[hh24:mi:ss], and $[yyyymmddhh24miss].

- yyyy: indicates a 4-digit year. The value is the year of the plan time.
- yy: indicates a two-digit year. The value is the last two digits of the year of the plan time.
- mm: indicates the month. The value is the month of the plan time.
- dd: indicates the day. The value is the day of the plan time.
- hh: indicates the 12-hour format. The value is the hour of the plan time.
- hh24: indicates the 24-hour format. The value is the hour of the plan time.
- mi: indicates the minute. The value is the minute of the plan time.
- ss: indicates the second. The value is the second of the plan time.

For details about how to obtain data N hours and minutes ago, see **Table 6-179**. This parameter cannot be used to obtain data N years and months ago using $[yyyy-N] or $[mm-N].

Table 6-179 Parameters for obtaining the plan time

Planned Time Scenario	Method
Next N Years	$[add_months(yyyymmdd,12*N)]
First N Years	$[add_months(yyyymmdd,-12*N)]
Last N Months	$[add_months(yyyymmdd,N)]
Last N Months	$[add_months(yyyymmdd,-N)]
N weeks before/after	$[yyyymmdd±7*N]
N days before/after	$[yyyymmdd±N]

Planned Time Scenario	Method
Before/After N Hours	You can obtain the time data in either of the following ways: ● $[hh24miss±N/24] ● $[User-defined time format ±N/24]. For example, to obtain the time format of the previous hour, run the following command: – Month: $[mm-1/24]. – Year: $[yyyy-1/24]. – Year and month: $[yyyymm-1/24]. – Obtain the year, month, and day: $[yyyymmdd-1/24]. – $[yyyymmdd-1-1/24]: indicates that the time of the previous day and the previous hour is used.
Before/After N minutes	You can obtain the time data in any of the following ways: ● $[hh24miss±N/24/60] ● $[yyyymmddhh24miss±N/24/60] ● $[mi±N/24/60] ● $[User-defined time format ±N/24/60] For example, to obtain the time format 15 minutes before the planned time, run the following command: – Year: $[yyyy-15/24/60] – Year and month: $[yyyymm-15/24/60] – Date: $[yyyymmdd-15/24/60] – Hour: $[hh24-15/24/60] – Minute: $[mi-15/24/60]

NOTE

- The replacement value of the scheduling parameter is determined when the instance is generated. Therefore, the replacement value of the scheduling parameter does not change with the actual running time of the instance.
- When the scheduling parameter is set to hour or minute, the parameter replacement value is determined by the planned scheduling time of the instance, that is, the planned scheduling time configured for the node scheduling. For example:
  - If the current node is a daily scheduling node and the planned scheduling time is 01:00, the value of Hour is 01.
  - If the current node is an hourly scheduling node, the planned scheduling time is set to 00:00-23:59, and the scheduling is performed every hour, the planned time of the first hourly instance is 00:00, and the value of the hour parameter is 00. The planned time of the second hourly instance is 01, and so on.

Service Parameters

The service ID is replaced with the actual ID of the current service, including the job ID and the instance ID generated by the job.

Table 6-180 Parameters for obtaining the service ID

Methods	Description
$job_id	Data Development Job ID For details about how to obtain the ID, see **Viewing Job Details**.
$instance_id	Job instance ID. (The instance ID is not generated during the test running of a single-node job and is not supported.) For details about how to obtain the ID, see **Viewing a Job Instance List**.

6.14 Usage Guidance

6.14.1 Referencing Parameters in Scripts and Jobs

This section describes how to reference parameters in scripts and jobs, application scope of the referenced parameters, and whether EL expressions and simple variable sets are supported, helping you better understand how to use workspace-level, script-level, and job-level parameters.

NOTE

Parameters can be set in environment variables, job parameters, and script parameters, but their application scopes are different. If there is a conflict when parameters in environment variables, job parameters, and script parameters of the same name, the calling priority is: **job parameters > environment variables > script parameters**.

Table 6-181 Methods of using parameters

Type	Scenario	Scope	Calling Method	Example
Environment variables/ constants	When configuring job parameters, you can extract a parameter that belongs to multiple jobs as an environment variable.	Current workspace	${*Environment variable*} ${*Environment constant*} For details about the configuration method, see **Environment Variable**.	EL expression: #{DateUtil.getDay(Job.planTime)} Simple variable set: ${yyyymmdd ±N}

Type	Scenario	Scope	Calling Method	Example
Job variables/ constants	Job parameters can be used in any node in jobs.	Current job	${*Job variable*} ${*Job constant*} For details about the configuration method, see **Configuring Job Parameters**.	EL expression: #{DateUtil.now()} Simple variable set: ${yyyymm±N}
Script parameters	Set the name and value of a custom field.	Current script	${*Script parameter*} For details about the configuration method, see **Script Parameter**.	Example script parameters: ${time} ${yyyymmddhh24miss} ${job_id} ${instance_id}

NOTE

Variables of an SQL script can be in ${} or ${dlf.} format. You can configure either type as needed. The configured variable format applies to SQL scripts, SQL statements in jobs, single-node jobs, and environment variables. For details about how to configure the script variable format, see **Configuring Script Variables**.

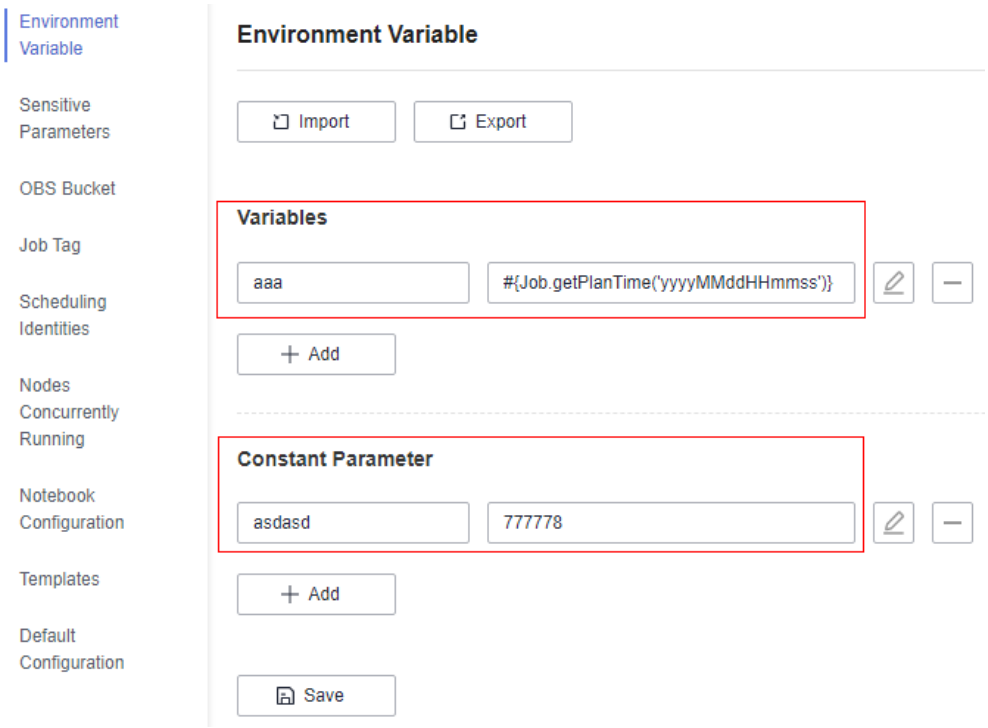
The default variable format is **${}**.

Environment Variable

Variables and constants can be defined in environment variables. Environment variables take effect in current workspace.

- The value of a variable (such as **workspace name**) varies depending on the workspace. When exporting a variable from a workspace and import it to another workspace, you must reconfigure its value.
- The value of a constant in different workspaces is the same. When importing a constant to another workspace, you do not need to reconfigure its value.

Figure 6-138 Environment Variable



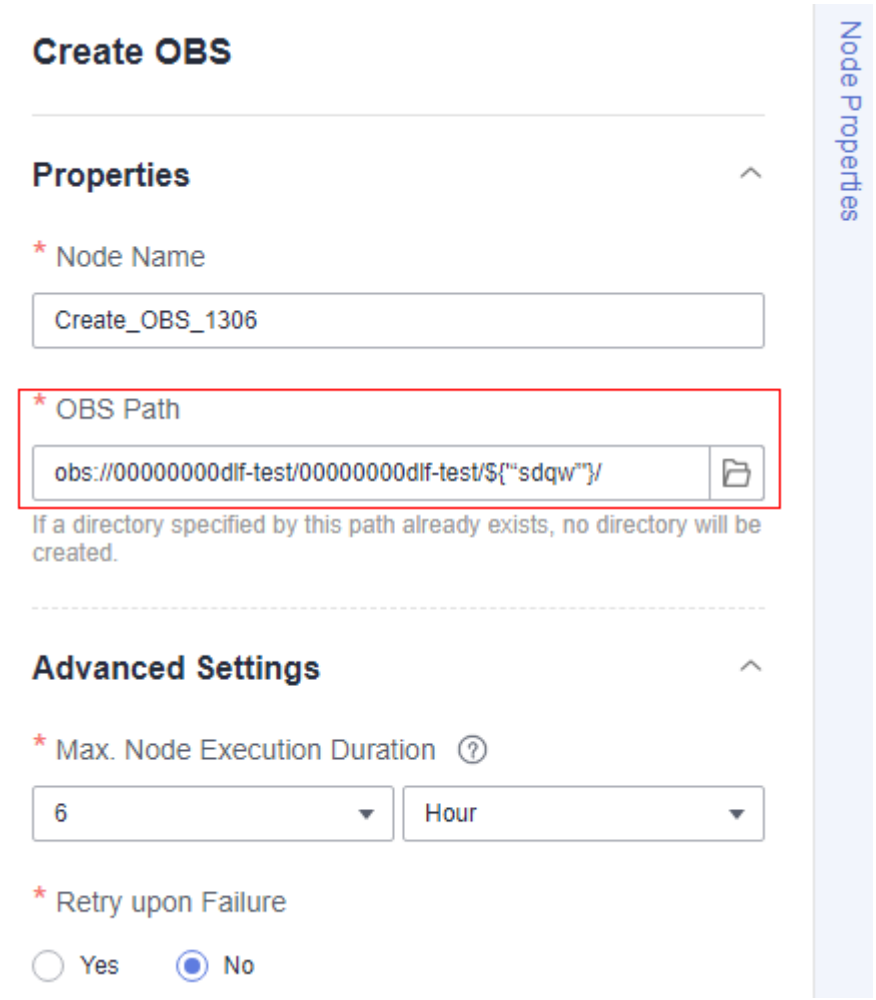
The specific application is as follows:

An environment variable has been added. The parameter name is **sdqw** and the parameter value is **wqewqewqe**.

Step 1 Open a created job and drag a **Create OBS** node from the node library to the canvas.

Step 2 On the **Node Properties** tab page, configure the node properties.

Figure 6-139 Create OBS



Step 3 Click **Save** and then **Monitor** to monitor the running status of the job.

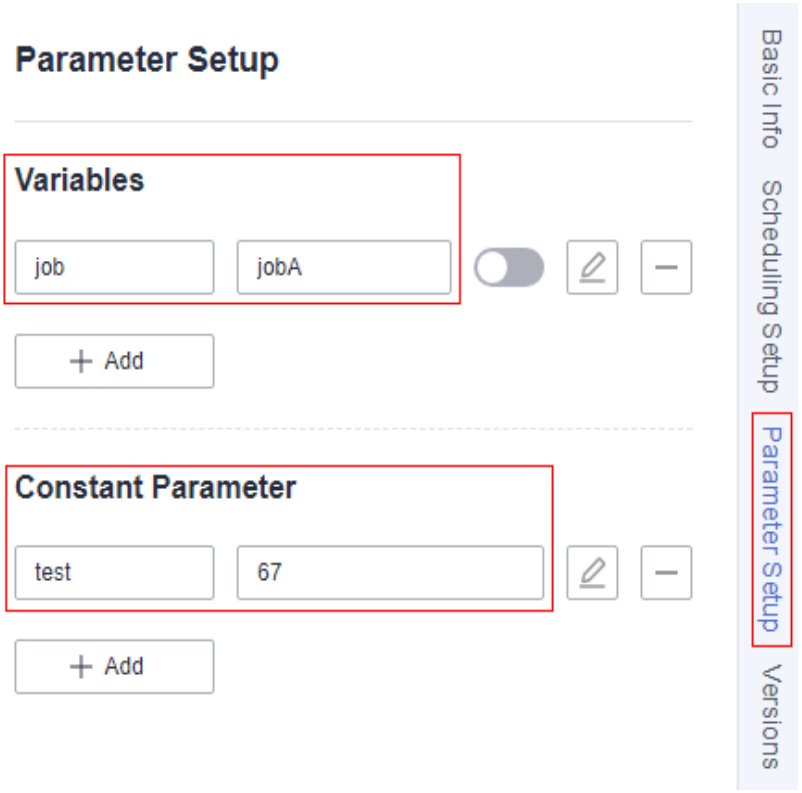
----End

Configuring Job Parameters

Parameters and constants can be defined in job parameters. Job parameters take effect in current job.

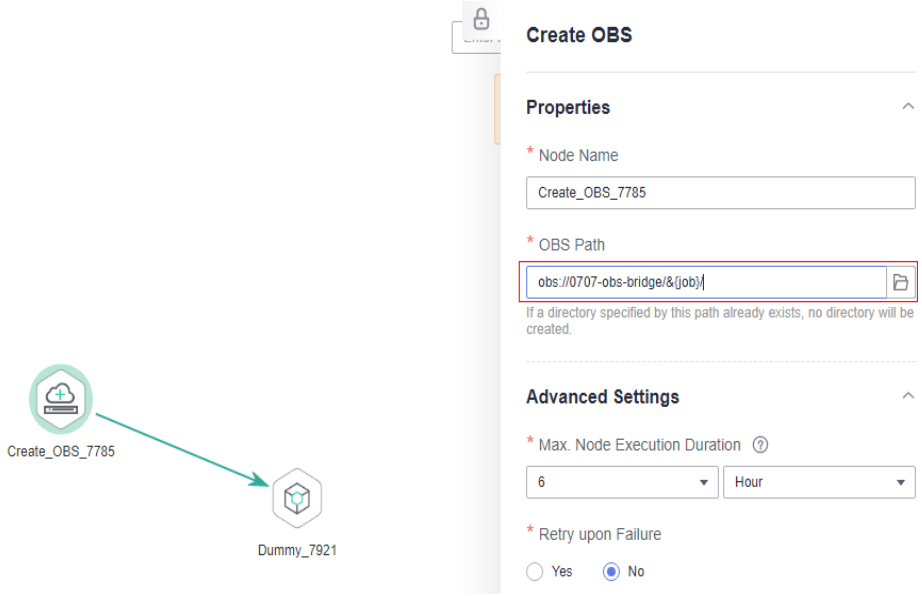
- The value of a parameter varies depending on the job. You need to reconfigure the parameter.
- The value of a constant in different jobs is the same. When importing a constant to another job, you do not need to reconfigure its value.

Figure 6-140 Job parameter.



After a job parameter is defined, it can be referenced by a job node.

Figure 6-141 Using a Job Parameter Configuration



Script Parameter

- Script parameters take effect in current script and it can be used in the following ways.

- For SQL scripts, you can directly enter parameters in the script editor (not supported for Flink SQL scripts). During job scheduling, you can assign values to parameters through node attributes, as shown in 2.
- For Shell scripts, you can enter a parameter and an interactive parameter in the upper part of the editor to transfer the parameters.
- Python scripts support parameter transfer.
- For SQL scripts, you can directly enter parameters in the script editor (not supported for Flink SQL scripts). When executing a script independently, you can configure parameters in the lower part of the editor shown in Figure 6-142.

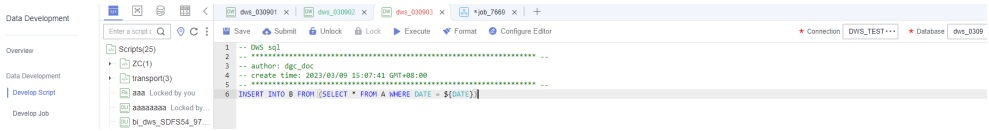
Figure 6-142 Configuring script parameters when executing a script independently

1. Developing a Python Script During script development, the script expression must contain variables. For example, if the variable in the SQL statement is DATE, set this parameter to ${DATE} in the script. In the job parameter configuration, you can compile the statement expression of the script parameter Date in 2.

   On the **script development** page, enter development statements in the editor, as shown in the following figure.

   ```
   INSERT INTO B FROM (SELECT * FROM A WHERE DATE = ${DATE})
   ```

   Figure 6-143 Developing a script

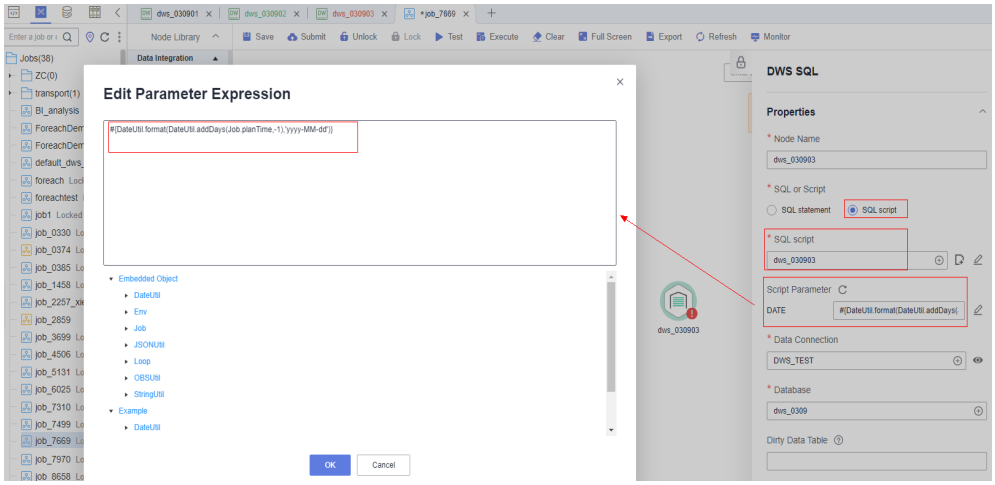
   

   After the dws_030903 script is compiled, save and submit the latest version of the script.

2. Develop batch jobs. When developing a job, you need to configure node attribute parameters.

   In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Job**.

Figure 6-144 Configuring script parameters when the script is executed by job scheduling



NOTE

- If the associated SQL script uses a parameter, the parameter name is displayed (**DATA** for example). Set the parameter value in the text box next to the parameter name. The parameter value can be **an EL expression**.
- If the associated SQL script or script parameters change, you can click to synchronize the changes or click to edit the changes.
- All nodes involving scripts, such as SQL scripts, shell scripts, and Python scripts, can use this method to reference script variables.

Simple Variable Set

The simple variable set provides a series of customized variables. Customized parameters are automatically replaced with specific values based on the service date, plan time, and parameter value format of task scheduling. In this way, parameters can be dynamically replaced during task scheduling. For details about the simple variable set, see **Simple Variable Set**.

6.14.2 Setting the Job Scheduling Time to the Last Day of Each Month

Scenario

When configuring job scheduling, you can set the scheduling time to the last day of each month using either of the two methods provided in the following table.

Table 6-182 Setting the scheduling time to the last day of each month

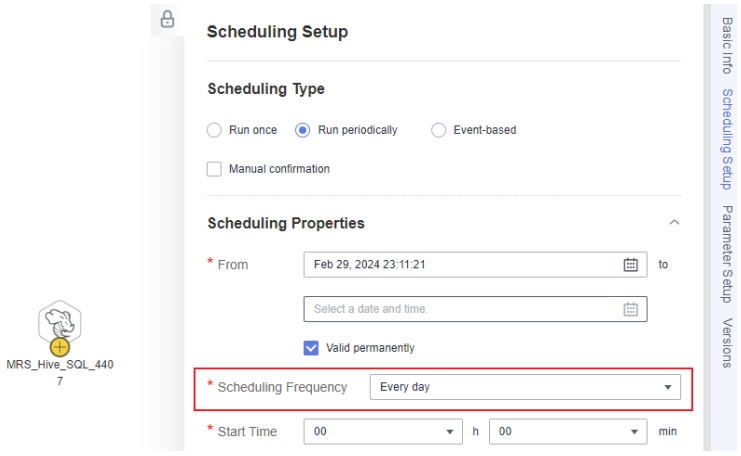
Method	Advantage	Procedure
Set the scheduling frequency to every day and use a condition expression to determine whether a day is the last day of each month.	This method applies to multiple scenarios. You can compile condition expressions to flexibly schedule jobs, for example, on the last day or 7th of each month.	**Method 1**
Set the scheduling frequency to every month and select the last day of each month.	You can set a specific job scheduling time instead of compiling any statements.	**Method 2**

Method 1

In DataArts Studio, create a job that is scheduled every day and add an empty Dummy node (which does not process data) to the job. You can set a condition expression on the connection line between the Dummy node and its subsequent node to check whether the current day is the last day of the current month. If it is the last day, the subsequent nodes are executed. Otherwise, the subsequent nodes are skipped.

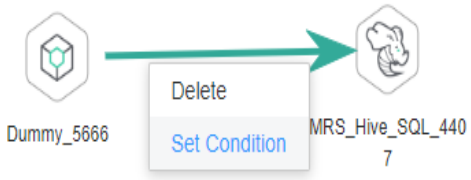
1. In the left navigation pane of the DataArts Factory console, choose **Data Development** > **Develop Job**.
2. Set **Scheduling Frequency** to **Every day**.

   Figure 6-145 Setting Scheduling Frequency to Every day



3. Right-click the connection line between the Dummy node and its subsequent node and select **Set Condition** to configure a condition expression that is used to determine whether to execute the subsequent node.

Figure 6-146 Configuring a condition expression

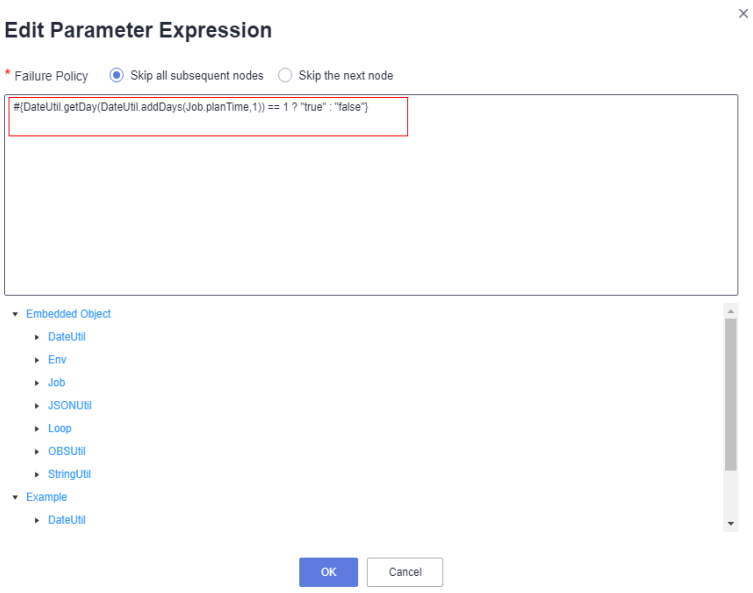


4. Configure the expression as follows:

```
#{DateUtil.getDay(DateUtil.addDays(Job.planTime,1)) == 1 ? "true" : "false"}
```

The expression is used to obtain the current time and check whether the next day is 1st of a month. If yes, the current day is the last day of the current month, and the subsequent node will be executed; if no, the subsequent node will be skipped.

Figure 6-147 Condition expression



For example, if you want a job to be executed on the last day and seventh day of each month, perform the following operation:

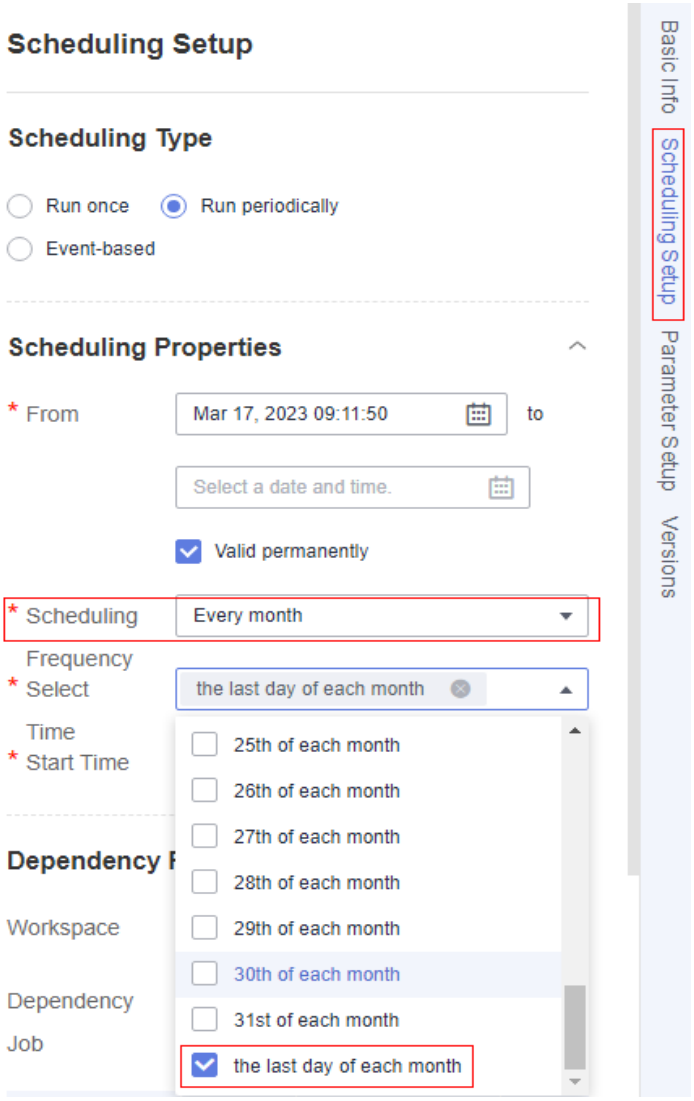
Configure the following expression to check whether the current day is 7th:

```
#{DateUtil.getDay(Job.planTime) == 7 ? "true" : "false"}
```

Method 2

1. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
2. Click **Scheduling Setup** on the right of the job canvas.
3. Set **Scheduling Type** to **Run periodically**, **Scheduling Frequency** to **Every month**, and **Select Time** to **the last day of each month**.

Figure 6-148 Setting the scheduling time to the last day of each month



After the scheduling time is configured, the job will be automatically executed on the last day of each month.

6.14.3 Configuring a Yearly Scheduled Job

Scenario

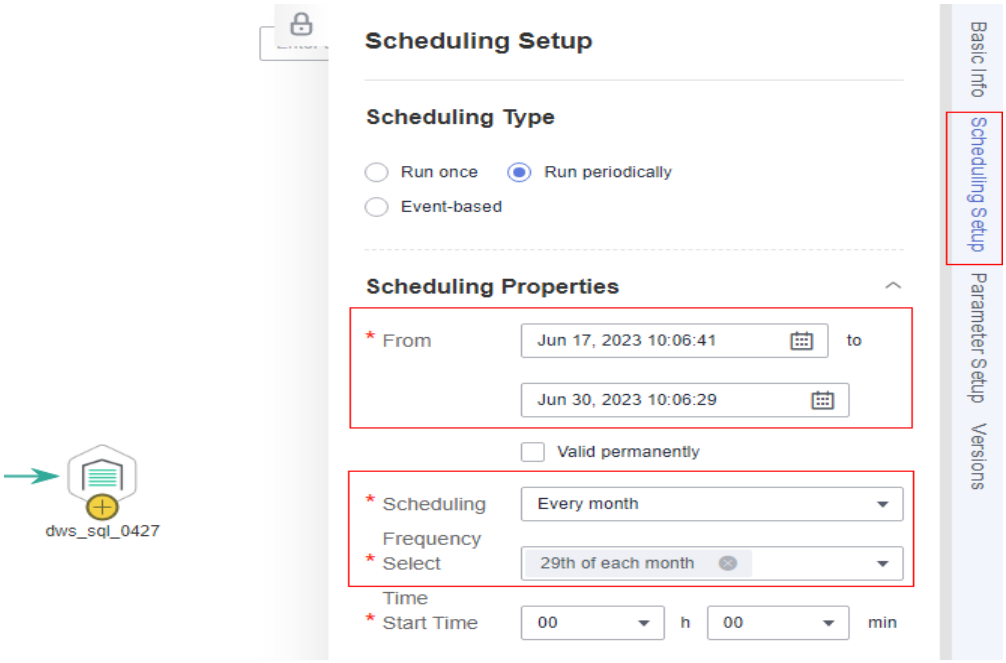
This section describes how to configure a job that is scheduled at a specified time of a year.

Procedure

In DataArts Studio, create a job that is scheduled every month and add an empty Dummy node (which does not process data) to the job. You can set a condition expression on the connection line between the Dummy node and its subsequent node to check whether the current time falls in the specified day (for example, June 29, 2023) for scheduling the job. If yes, the subsequent node is executed. Otherwise, the subsequent node is skipped.

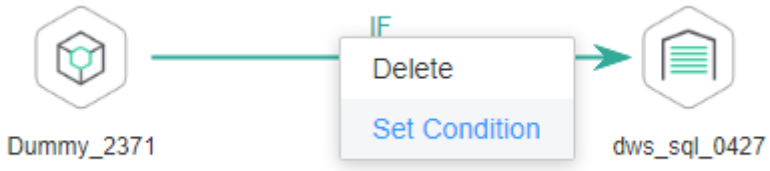
1. In the left navigation pane of the DataArts Factory console, choose **Data Development** > **Develop Job**.
2. Set **Scheduling Frequency** to **Every month**.

   Figure 6-149 Setting Scheduling Frequency to Every month



3. Right-click the connection line between the Dummy node and its subsequent node and select **Set Condition** to configure a condition expression that is used to determine whether to execute the subsequent node.

   Figure 6-150 Configuring a condition expression

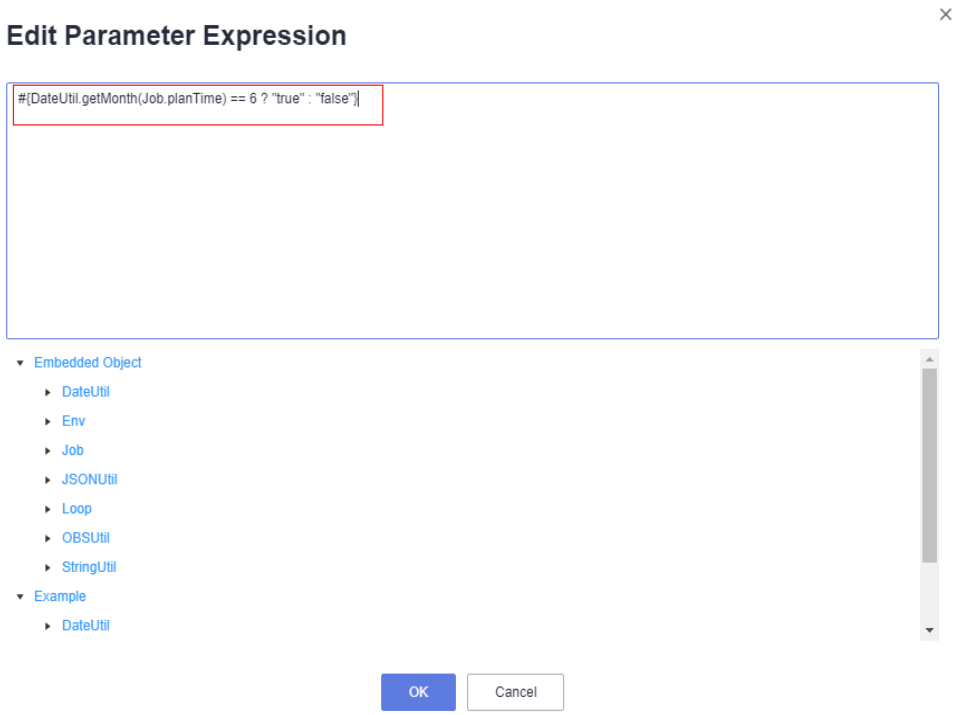


4. Configure the expression as follows:

```
#{DateUtil.getMonth(Job.planTime) == 6 ? "true" : "false"}
```

   The expression is used to obtain the current time and check whether it falls in June. If yes, the subsequent node will be executed; if no, the subsequent node will be skipped.

Figure 6-151 Condition expression



6.14.4 Using PatchData

Scenario

In the migration of a project, if you want to supplement historical business data in a previous period and view details of the historical data, PatchData can meet your requirements.

A job executes a scheduling task to generate a series of instances in a certain period of time. This series of instances are called PatchData. PatchData can be used to fix the job instances that have data errors in the historical records or to build job records for debugging programs.

NOTE

- In addition to SQL scripts, PatchData supports other nodes.
- If the content of a SQL script changes, the PatchData job runs the latest script.
- When you use PatchData, if the variable in the SQL statement is **DATE**, enter **${DATE}** in the script. The script parameter **DATE** is then automatically added to the job parameters, and its value can be an EL expression. If the variable is a time variable, enter the expression of the **DateUtil** embedded object. The platform automatically converts the expression into a historical date. For details about how to use EL expressions, see **EL Expressions**.
- PatchData jobs support script parameters and global environment variables as well as job parameters.

Constraints

- PatchData is available only when periodic scheduling is configured for the data development job.

Example

Scenario

Among the product data tables of a company, there is a source data table A that records the product sales amount. To import the historical product sales amount to the destination table B, you can create a PatchData job.

Table 1 lists the source and destination tables.

Table 6-183 Source and destination tables

Source Table	Destination Table
A	B

Procedure

1. Prepare the source and destination tables. To facilitate subsequent job execution and verification, you need to create a source DWS table and a destination DWS table and insert data into the tables.
   a. Create a DWS table. You can create a DWS SQL script on the DataArts Factory console of DataArts Studio and run the following SQL statements:
   ```
   /* Create tables. */
   CREATE TABLE A (PRODUCT_ID INT, SALES INT, DATE DATE);
   CREATE TABLE B (PRODUCT_ID INT, SALES INT, DATE DATE);
   ```
   b. Insert sample data into the source data table. You can create a DWS SQL script on the DataArts Factory console of DataArts Studio and run the following SQL statements:
   ```
   /* Insert sample historical data into the source table. */
   INSERT INTO A VALUES ('1','60', '2022-03-01');
   INSERT INTO A VALUES ('2','80', '2022-03-01');
   INSERT INTO A VALUES ('1','50', '2022-02-28');
   INSERT INTO A VALUES ('2','55', '2022-02-28');
   INSERT INTO A VALUES ('1','60', '2022-02-27');
   INSERT INTO A VALUES ('2','45', '2022-02-27');
   ```
2. Develop a PatchData script. Ensure that the script expression contains a time variable. (For example, if the variable in the SQL statement is **DATE**, enter **${DATE}** in the script.) You can set the expression for script parameter **DATE** in job parameter settings in 3.

   On the **Develop Script** page, enter following statement in the editor:
   ```
   INSERT INTO B (SELECT * FROM A WHERE DATE = ${DATE})
   ```

   Figure 6-152 Developing a script

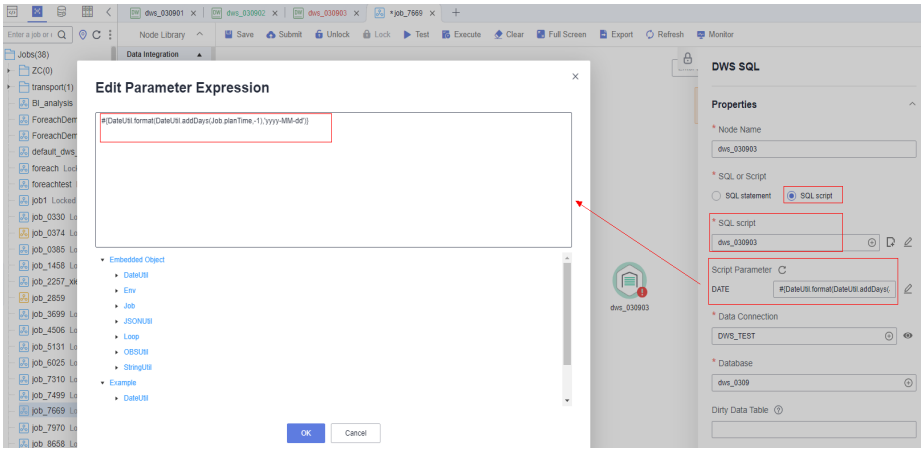
   ```
   -- DWS sql
   -- ********************************************************************** --
   -- author: [redacted]
   -- create time: 2023/05/23 17:03:02 GMT+08:00
   -- ********************************************************************** --
   INSERT INTO B (SELECT * FROM A WHERE DATE = ${DATE})
   ```


After compiling the script, save it and submit the latest version.

3. Develop a PatchData batch processing job. When developing the job, you need to configure the node attributes and scheduling period.

   In the left navigation pane of the DataArts Factory console, choose **Data Development** > **Develop Job**.

Figure 6-153 Node parameters



NOTE

- If the job-associated SQL script uses a parameter, the parameter name (such as **DATE**) is displayed. Set the parameter value in the text box next to the parameter name. The parameter value can be an EL expression. For details about EL expressions, see **Expression Overview**.

  If the parameter is time, view the example expression of the DateUtil embedded object. The platform automatically replaces the parameter with the historical date of the patch data (determined by the service date of the patch data).

  You can also directly enter a SQL expression.

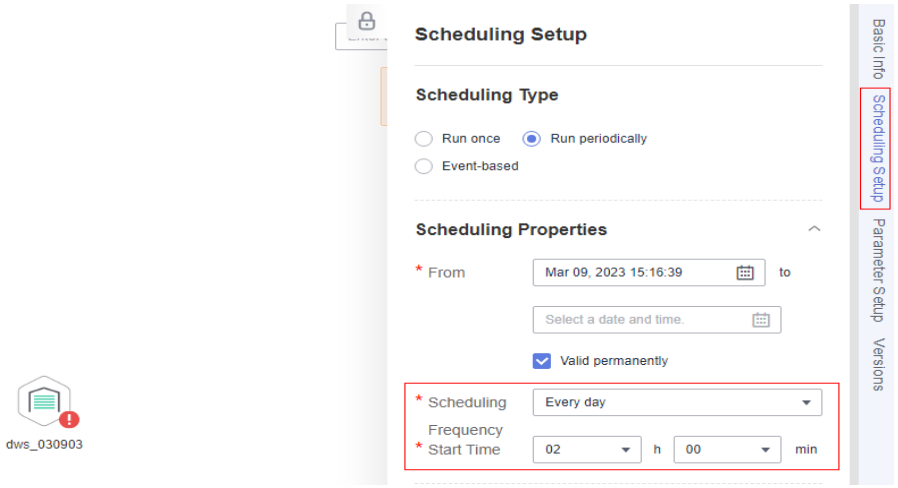
- If the parameters of the associated SQL script change, you can click to synchronize the change or click to edit the parameters.
- The following is an example of script parameters:

  Example: #{DateUtil.format(DateUtil.addDays(Job.planTime,-1),'yyyy-MM-dd')}

  - **Job.planTime** indicates the planned job time, and *yyyy-MM-dd* indicates the time format.
  - If the planned job time is March 2, the previous day is March 1. The planned job time will be replaced by the configured patch data service date.
  - The **Job.planTime** is converted into a time in the *yyyy-MM-dd* format using an expression.

Configure the scheduling period of the PatchData job. Click **Scheduling Setup** and set **Scheduling Frequency** to **Every day**.

Figure 6-154 Configuring the scheduling period



NOTE

- If **Scheduling Frequency** is set to **Every day**, the job is scheduled every day, and a PatchData instance is generated. You can view the statuses of PatchData instances on the **Monitor Instance** page. On the **Monitor Instance** page, view the instance information about the job and perform more operations on instances as required.
- The job scheduling time takes effect from March 9, 2023, and the job is scheduled at 02:00 every day.
- Run the following SQL statement to check whether destination table B contains data of source table A:

  SELECT * FROM B

After configuring the parameters, save and submit the latest version of the job and test the job.

Click **Execute** to run the job.

4. Create a PatchData task.

   After creating a periodic job, you need to configure PatchData for the job.

   a. In the left navigation pane of DataArts Factory, choose **Monitoring** > **Monitor Job**.

   b. Click the **Batch Job Monitoring** tab. In the **Operation** column of the job, choose **More** > **Configure PatchData**. The **Configure PatchData** page is displayed.

      If you want to supplement historical data from February 27, 2023 to March 1, 2023, set **Date** to **Feb 28, 2023 00:00:00 – Mar 02, 2023 23:59:59**. The system automatically transfers the configured date to the planned job time. In the expression of the script time variable **DATE**, the defined time is the planned job time minus one day. That is, the time of the day before the planned job time is the time range (**Feb 27, 2023 to Mar 1, 2023**) for PatchData.

Figure 6-155 Configuring PatchData

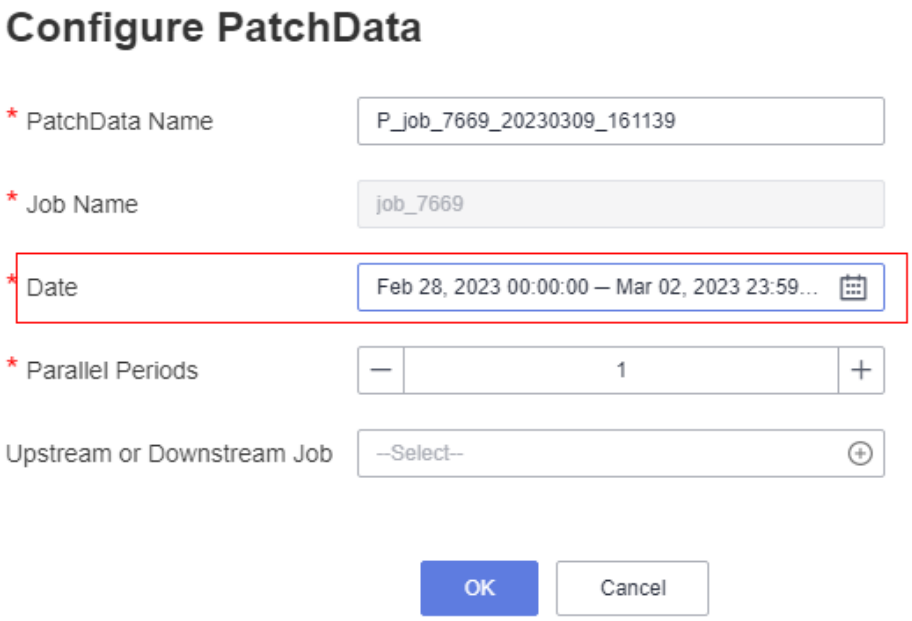


Table 6-184 Description

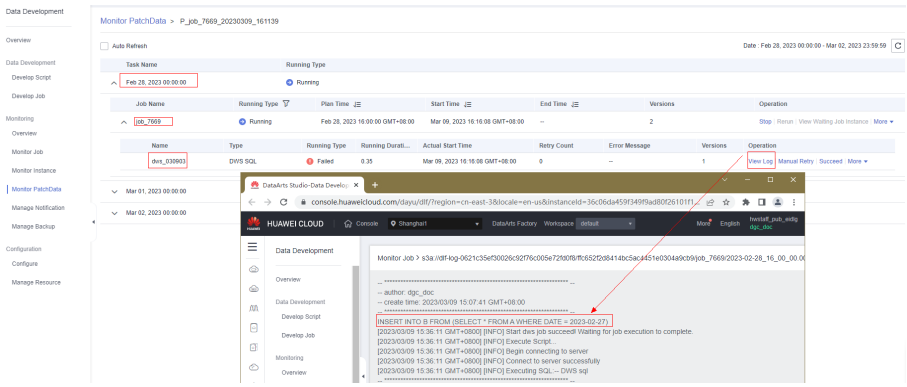
Parameter	Description
PatchData Name	Name of the automatically generated PatchData task. The value can be modified.
Job Name	Name of the job that requires PatchData, which is automatically displayed
Date	Period of time when PatchData is required. This date is transferred to the planned job time. When the job is executed, the planned job time is replaced by the time in the PatchData. **NOTE** PatchData can be configured for a job multiple times. However, avoid configuring PatchData multiple times on the same date to prevent data duplication or disorder. If you select **Patch data in reverse order of date**, the patch data of each day is in positive sequence. **NOTE** • This function is applicable when the data of each day is not coupled with each other. • The PatchData job will ignore the dependencies between the job instances created before this date.

Parameter	Description
Parallel Periods	Number of instances to be executed at the same time. A maximum of five instances can be executed at the same time. **NOTE** Set this parameter based on the site requirements. For example, if a CDM job instance is used, data cannot be supplemented at the same time. The value of this parameter can only be set to **1**.
Upstream or Downstream Job	This parameter is optional. Select the downstream jobs (jobs that depend on the current job) that require PatchData. You can select multiple jobs.

c. Click **OK**. The system starts to run the PatchData task based on the configured scheduling period.

d. On the **Monitor PatchData** page, you can view the PatchData task status, date, number of parallel periods, PatchData job name, and stopped tasks. You can also view details about the PatchData task.

Figure 6-156 Querying PatchData details



e. Run the following SQL statement to check whether destination table B contains historical data of source table A:

```
SELECT * FROM B
```

6.14.5 Obtaining the Output of an SQL Node

This section describes how to obtain the output of an SQL node and apply the output to subsequent nodes or judgment in job development.

Scenario

When you use EL expression **#{Job.getNodeOutput("Name of the previous node")}** to obtain the output of the previous node, the output is a two-dimensional array, for example, **[["Dean",...,"08"],...,["Smith",...,"53"]]**. To obtain the values in the array, use either of the methods provided in **Table 6-185**.

Table 6-185 Methods for obtaining output values

Method	Key Configuration	Application Scenario Requirements
Obtaining Output Value Using StringUtil	If the output of the SQL node contains only one field, for example **[["11"]]**, you can use the StringUtil EL expression with an embedded object to split the two-dimensional array and obtain the field value in the output of the previous node. #{StringUtil.split(StringUtil.split(StringUtil.split(Job.getNodeOutput("***Name of the previous node***"),"]")[0],"[")[0],"\\"")[0]}	This method is easy to use but has the following requirements on application scenarios: • The output of the previous SQL node contains only one field, for example, **[["11"]]**. • The output value is a string. The application scenario must support the string data type. For example, if the IF condition needs to be used to judge the size of the output value, the string type is not supported. In this case, this method cannot be used.
Obtaining Output Values Using the For Each Node	Use the For Each node to cyclically obtain the values in the two-dimensional array in the dataset. • For Each node dataset: #{Job.getNodeOutput('***Name of the previous node***')} • Subjob parameters of the For Each node: #{Loop.current[***Index***]}	This method is applicable to more scenarios, though jobs need to be split into main jobs and subjobs.

Obtaining Output Value Using StringUtil

Scenario

The StringUtil EL expression with an embedded object is used to split the two-dimensional array result and obtain the output field value of the previous node, which is a string.

In this example, the MRS Hive SQL node returns a two-dimensional array that contains a single field. The data sent by the Kafka Client node is defined as the StringUtil EL expression with an embedded object. You can use this expression to split the two-dimensional array and obtain the output field value of the MRS Hive SQL node.

NOTE

To make it easy to view the obtained value, this example uses the Kafka Client node. In practice, you can select a subsequent node type as needed. By using a StringUtil EL expression with an embedded object on the node, you can obtain the data value returned by the previous node.

Figure 6-157 Example job

The key configuration of the Kafka Client node is the **Sent Content** parameter. Set it as follows:

```
#{StringUtil.split(StringUtil.split(StringUtil.split(Job.getNodeOutput("count95"),"]")[0],"[")[0],"\\"")[0]}
```

Configuration Method

Step 1 Log in to the DataArts Studio console, locate the target DataArts Studio instance, and click **Access** on the instance card.

Step 2 Click the **Workspaces** tab. In the workspace list, locate the target workspace and click **DataArts Factory**.

Step 3 Create table **student_score**. Create a temporary Hive SQL script, select a Hive connection and database, paste the following SQL statement, and run the script. After the script is successfully executed, delete it.

```
CREATE TABLE `student_score` (`name` String COMMENT '', `score` INT COMMENT '');
INSERT INTO
    student_score
VALUES
    ('ZHAO', '90'),
    ('QIAN', '88'),
    ('SUN', '93'),
    ('LI', '94'),
    ('ZHOU', '85'),
    ('WU', '79'),
    ('ZHENG', '87'),
    ('WANG', '97'),
    ('FENG', '83'),
    ('CEHN', '99');
```

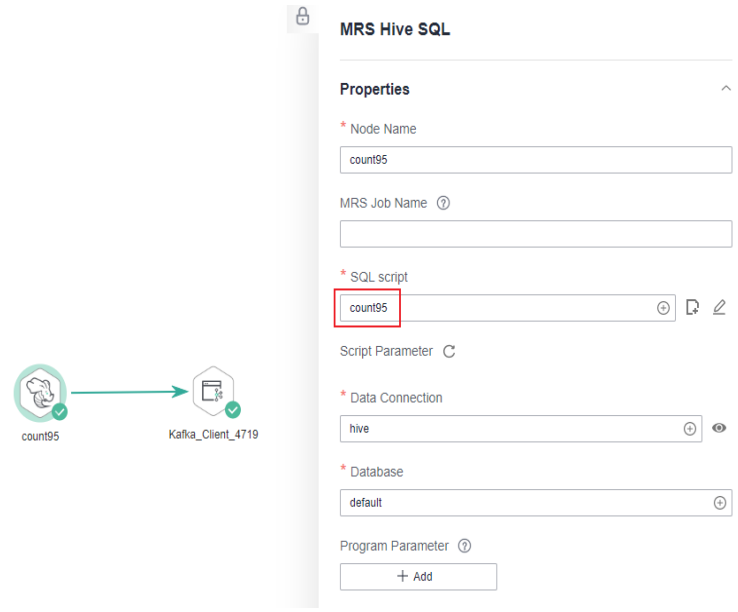
Step 4 Create the Hive SQL script to be invoked by the MRS Hive SQL node. Create a Hive SQL script named **count95**, select a Hive connection and database, paste the following SQL statement, and submit a version.

```
--Obtain the number of students whose scores are higher than 95 from the student_score table.--
SELECT count(*) FROM student_score WHERE score> "95" ;
```

Step 5 On the **Develop Job** page, create a data development job. Drag an MRS Hive SQL node and a Kafka Client node and drop them on the canvas. Click and hold to connect the nodes, as shown in **Figure 6-157**.

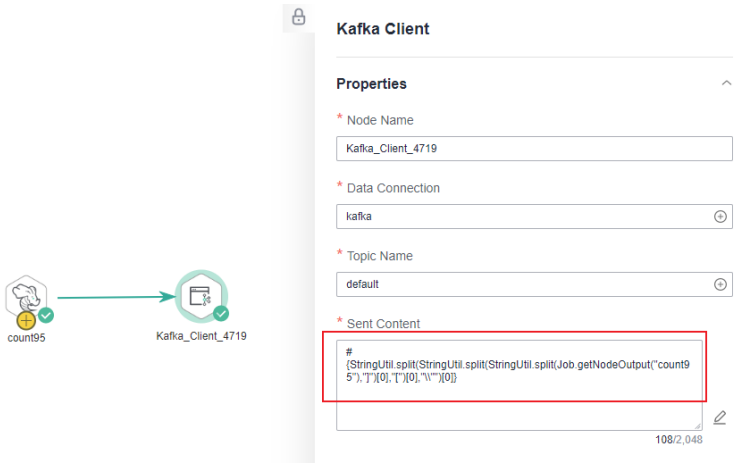
Step 6 Configuring parameters for an MRS Hive SQL node Select the **count95** script submitted in **Step 4** for **SQL script** and select a Hive connection and database.

Figure 6-158 Configuring parameters for an MRS Hive SQL node



Step 7 Configure parameters for the Kafka Client node. Set **Sent Content** to **#{StringUtil.split(StringUtil.split(StringUtil.split(Job.getNodeOutput("count95"),"]")[0],"[")[0],"\\"")[0]}** and select a Kafka connection and a topic name.

Figure 6-159 Configuring parameters for the Kafka Client node



Step 8 After the node configuration is complete, click **Test**. After the job test is successful, right-click the Kafka Client node to view its log. You can find that the two-dimensional array **[["2"]]** returned by the MRS Hive SQL node has been converted to **2**.

NOTE

You can set **Sent Content** of the Kafka Client node to **#{Job.getNodeOutput("count95")}** and run the job. Then you can view the log of the Kafka Client node to verify that the result returned by the MRS Hive SQL node is two-dimensional array **[["2"]]**.

Figure 6-160 Check the Kafka Client node logs.

----End

Obtaining Output Values Using the For Each Node

Scenario

You can use the For Each node and the EL expression #{Loop.current[0]} with a Loop embedded object to cyclically obtain the output values of the previous node.

In this example, the MRS Hive SQL node returns a two-dimensional array that contains multiple fields. You can use the For Each node which cyclically invokes the subjobs of the Kafka Client node and set **Sent Content** of the Kafka Client node to **#{Loop.current[]}** to obtain the output values of the MRS Hive SQL node.

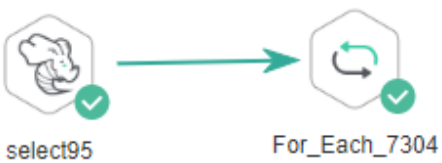
NOTE

To make it easy to view the obtained values, this example uses the Kafka Client node as the subjob node of the For Each node. In practice, you can select a subjob node type as needed. By using an EL expression with an embedded Loop object on the node, you can obtain the values returned by the previous node of the For Each node.

Orchestrate the main job shown in Figure 6-161. Key configurations of the For Each node are as follows:

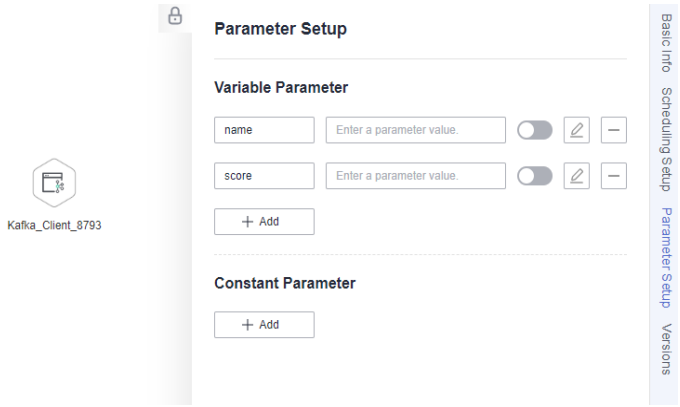
- **Dataset**: Enter the execution result of the select statement on the Hive SQL node. Use the **#{Job.getNodeOutput("select95")}** expression, where **select95** is the name of the previous node.
- **Subjob Parameter Name**: Enter the parameter name defined in the subjob. Transfer the parameter value defined in the main job to the subjob. Set the subjob parameter names to **name** and **score**, whose values are those in the first and second columns in the dataset, respectively. EL expressions **#{Loop.current[0]}** and **#{Loop.current[1]}** are used.

Figure 6-161 Example main job



For the subjobs selected for the For Each node, you must set their parameter names so that the main job can identify the parameter definitions.

Figure 6-162 Example subjob



Configuration Method

Developing a Subjob

Step 1 Log in to the DataArts Studio console, locate the target DataArts Studio instance, and click **Access** on the instance card.

Step 2 Click the **Workspaces** tab. In the workspace list, locate the target workspace and click **DataArts Factory**.

Step 3 On the **Develop Job** page, create a data development subjob named **EL_test_slave**. Select a Kafka Client node, configure job parameters, and orchestrate the job shown in **Figure 6-162**.

Set the parameter name to **name** and **score**. This parameter is only used by the For Each node in the main job to identify subjob parameters. You do not need to set the parameter value.

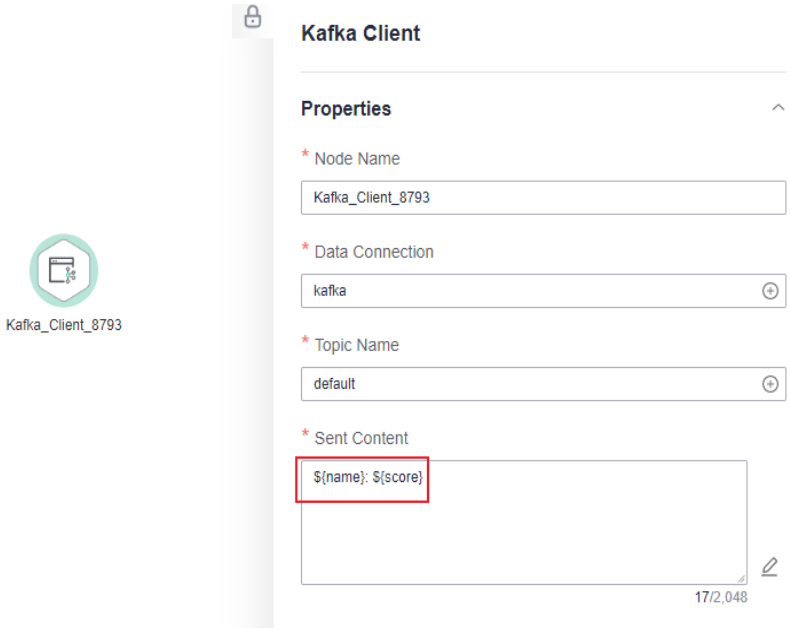
Step 4 Configure parameters for the Kafka Client node. Set **Sent Content** to **${name}: ${score}** and select a Kafka connection and a topic name.

NOTE

Do not use the **#{Job.getParam("job_param_name")}** EL expression because this expression can only obtain the values of the parameters configured in the current job, but cannot obtain the parameter values transferred from the parent job or the global variables configured in the workspace. The expression only works for the current job.

To obtain the parameter values passed from the parent job and the global variables configured for the workspace, you are advised to use the **${job_param_name}** expression.

Figure 6-163 Configuring parameters for the Kafka Client node



Step 5 Submit the subjob after the configuration is complete.

----End

Developing a Main Job

Step 1 Go to the **Develop Script** page.

Step 2 Create table **student_score**. Create a temporary Hive SQL script, select a Hive connection and database, paste the following SQL statement, and run the script. After the script is successfully executed, delete it.

```
CREATE TABLE `student_score` (`name` String COMMENT '', `score` INT COMMENT '');
INSERT INTO
   student_score
VALUES
   ('ZHAO', '90'),
   ('QIAN', '88'),
   ('SUN', '93'),
   ('LI', '94'),
   ('ZHOU', '85'),
   ('WU', '79'),
   ('ZHENG', '87'),
   ('WANG', '97'),
   ('FENG', '83'),
   ('CEHN', '99');
```

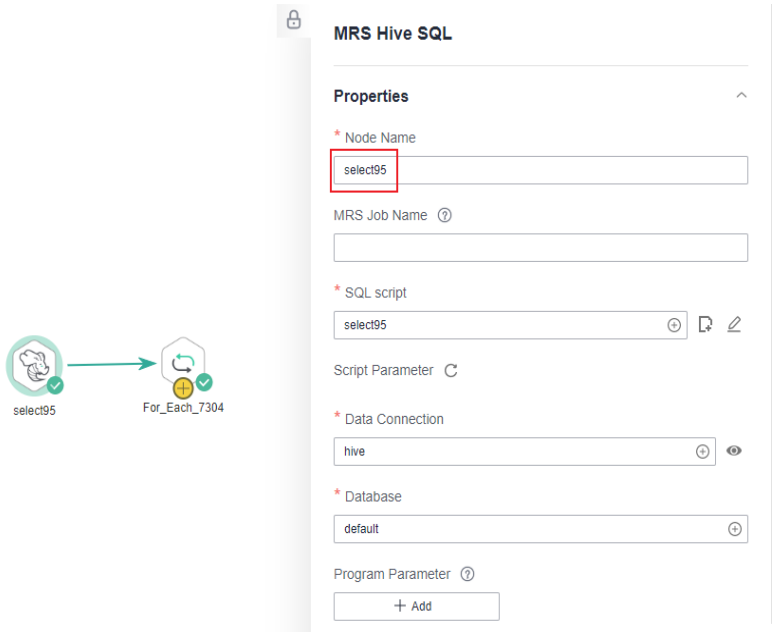
Step 3 Create the Hive SQL script to be invoked by the MRS Hive SQL node. Create a Hive SQL script named **select95**, select a Hive connection and database, paste the following SQL statement, and submit a version.

```
--Display the names and scores of students whose scores are higher than 95 in the student_score table.--
SELECT * FROM student_score WHERE score> "95" ;
```

Step 4 On the **Develop Job** page, create a data development job named **EL_test_master**. Drag a HIVE SQL node and a For Each node and drop them on the canvas. Click and hold to connect the nodes, as shown in **Figure 6-161**.

Step 5 Configure parameters for the MRS Hive SQL node. Select the **select95** script submitted in **Step 3** for **SQL script** and select a Hive connection and database.

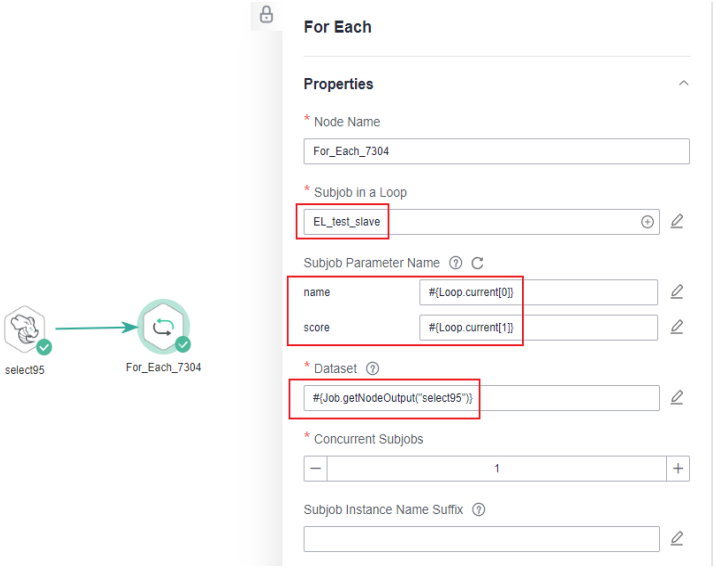
Figure 6-164 Configuring parameters for an MRS Hive SQL node



Step 6 Configure properties for the For Each node.

- **Subjob in a Loop**: Select **EL_test_slave**, the subjob that has been developed.
- **Dataset**: Enter the execution result of the select statement on the Hive SQL node. Use the **#{Job.getNodeOutput("select95")}** expression, where **select95** is the name of the previous node.
- **Subjob Parameter Name**: Enter the parameter name defined in the subjob. Transfer the parameter value defined in the main job to the subjob. Set the subjob parameter names to **name** and **score**, whose values are those in the first and second columns in the dataset, respectively. EL expressions **#{Loop.current[0]}** and **#{Loop.current[1]}** are used.

Figure 6-165 Configuring properties for the For Each node



Step 7 Save the job.

----End

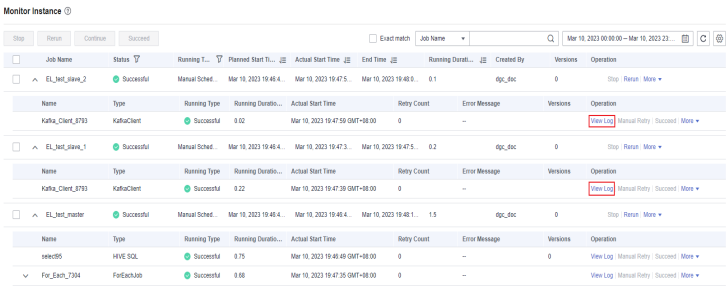
Testing the Main Job

Step 1 Click **Test** above the main job **EL_test_master** canvas to test the job. After the main job is executed, the subjob **EL_test_slave** is cyclically invoked through the For Each node and executed.

Step 2 In the navigation pane on the left, choose **Monitor Instance** to view the job execution result.

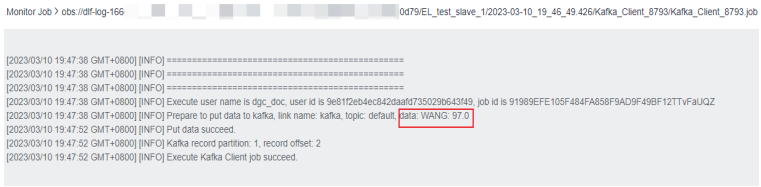
Step 3 After the job is executed, view the cyclic execution result of the subjob **EL_test_slave** on the **Monitor Instance** page.

Figure 6-166 Execution result of the subjob



Step 4 View the log of the cyclic execution of subjob **EL_test_slave**. The log shows that the output values of the previous node of the For Each node was obtained through the For Each node and the EL expression with a Loop embedded object.

Figure 6-167 Viewing the log



----End

6.14.6 Obtaining the Maximum Value and Transferring It to a CDM Job Using a Query SQL Statement

Scenario

You can run a query SQL statement to transfer the obtained maximum time value to a CDM job. In the advanced attributes of the CDM job, the where clause is used to determine the maximum time range to obtain the data to be migrated and complete the incremental data migration.

Constraints

1. You have completed operations in **Creating a Data Connection**.
2. You have completed operations in **Creating a Database**.

Examples

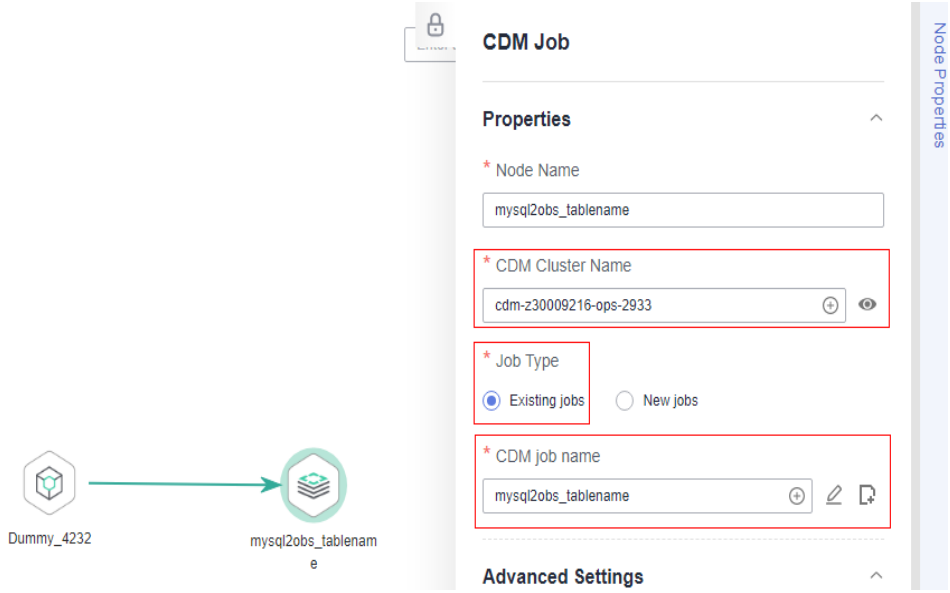
Creating an SQL Script

1. In the left navigation pane of DataArts Factory, choose **Data Development** > **Develop Script**.
2. Create an SQL script. This section uses the MRS Spark SQL script as an example.
3. Select a created data connection and database.
4. Compile the SQL script to obtain the maximum time data from table1.
   ```
   select max(time) from table1
   ```
5. Save and submit the version. The **maxtime** script is created.

Creating a Pipeline Subjob

1. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
2. Select a CDM Job node and configure the node properties.

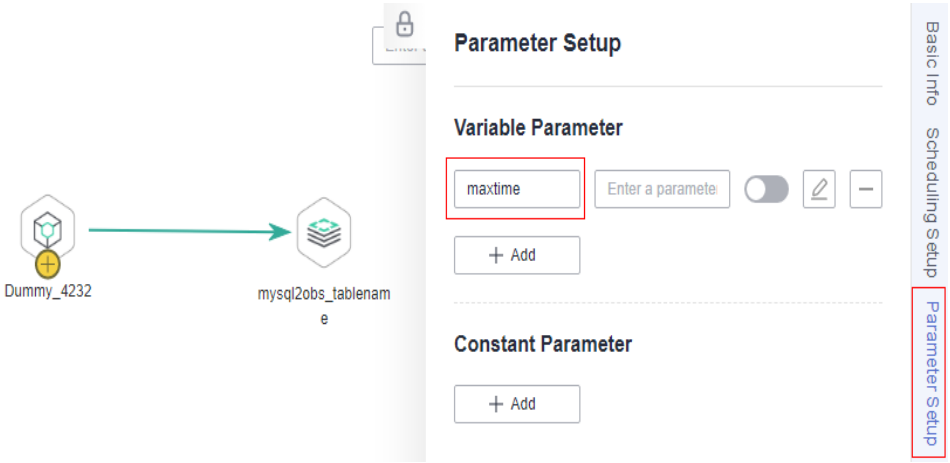
Figure 6-168 Configuring CDM Job node properties



Select a CDM cluster and associate the node with an existing CDM job.

Configure the job parameters and add job parameter **maxtime**.

Figure 6-169 Configuring job parameters

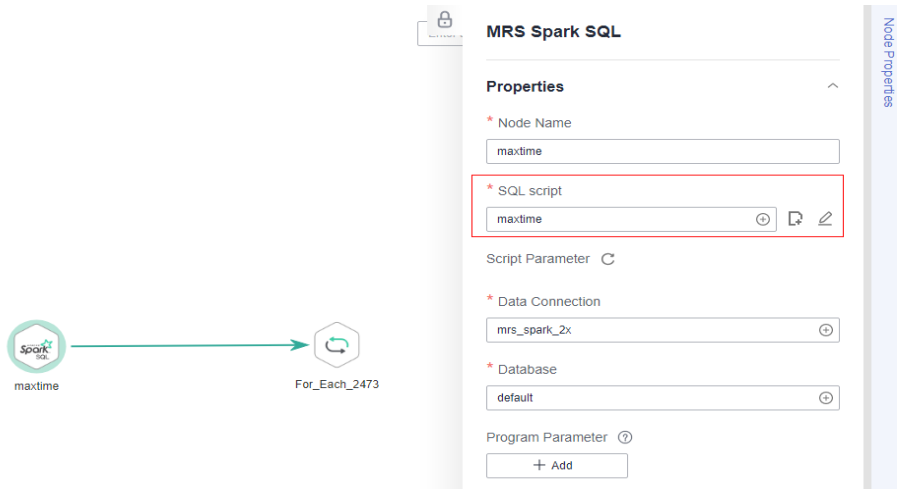


3. Save and submit the version. The subjob **sub** is created.

Creating a Pipeline Job

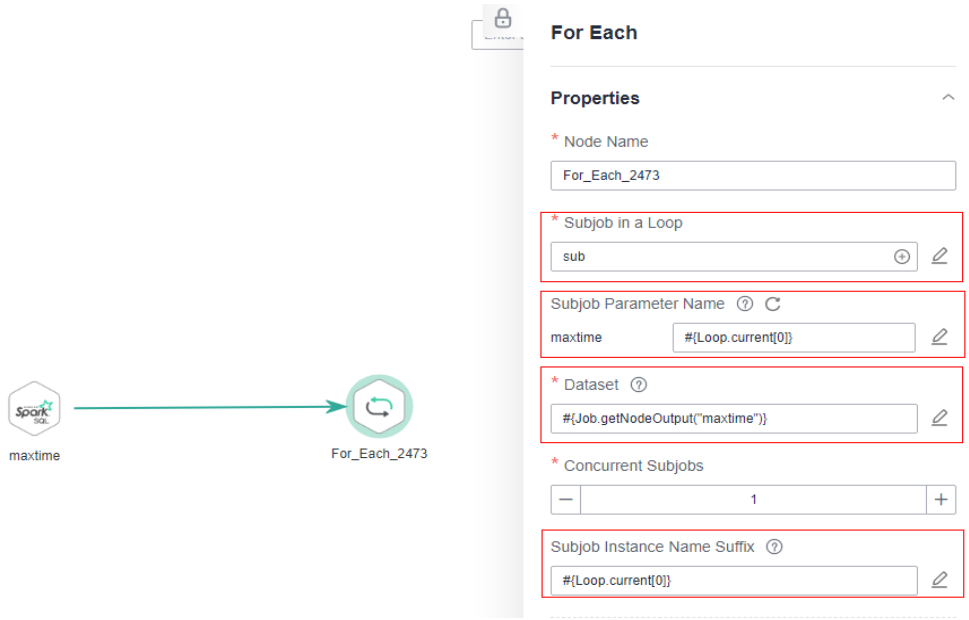
1. In the left navigation pane of DataArts Factory, choose **Development** > **Develop Job**.
2. Select an MRS Spark SQL node and a For Each node to execute the CDM subjob cyclically.
3. Configure properties of the MRS Spark SQL node and associate the node with the created **maxtime** script.

Figure 6-170 Configuring properties for the MRS Spark SQL node



4. Configure properties of the For Each node and associate the node with the created CDM subjob.

Figure 6-171 Configuring properties for the For Each node



After associating the node with the created subjob **sub**, write a parameter expression.

```
#{Loop.current[0]}
```

Configure the data set, with an EL expression supported.

```
#{Job.getNodeOutput("maxtime")}
```

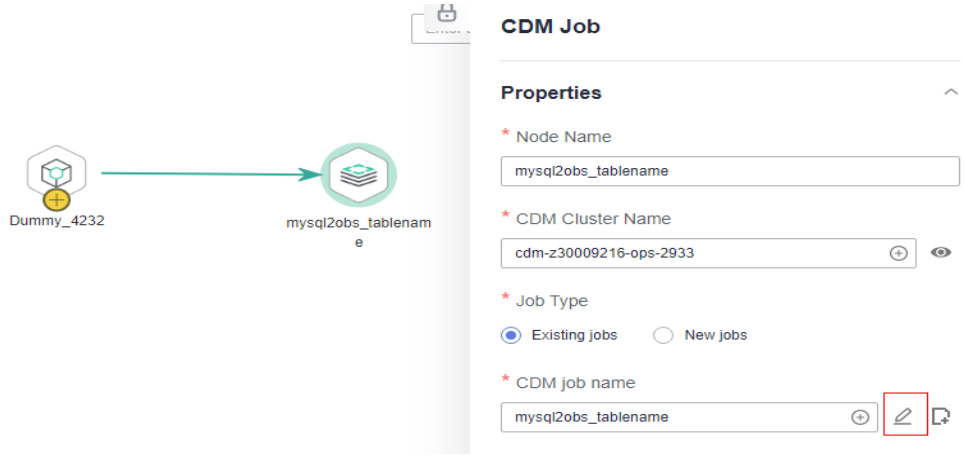
5. Save and submit the version. The job is created.

Obtaining the Maximum Time Value from the CDM Job Using a Where Clause and Transferring the Value to the Destination Job

1. Open the created subjob.

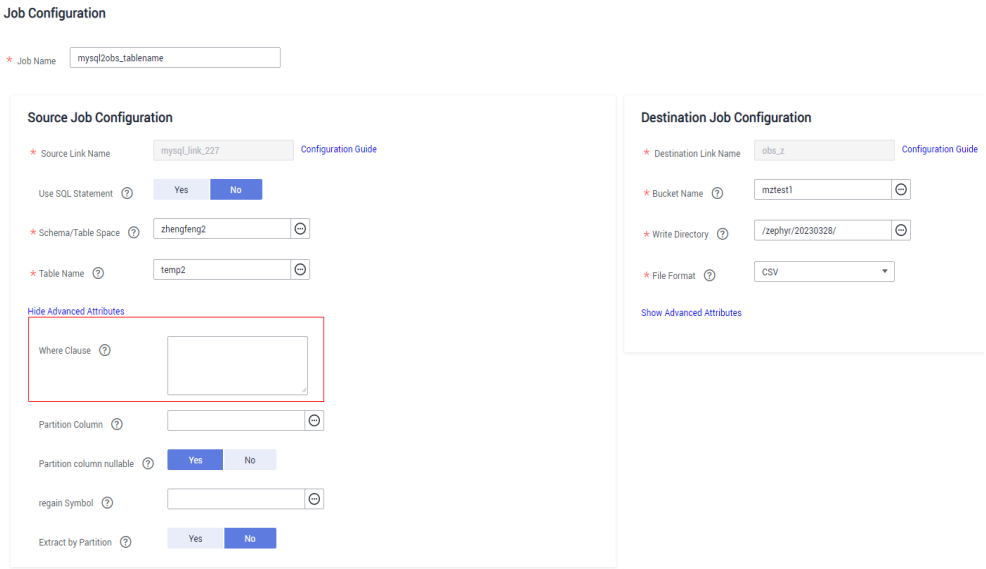
2. Click next to the job name to go to the job configuration page.

Figure 6-172 Editing the CDM job



3. In the advanced attributes of the source job configuration, configure a where clause to obtain the data to be migrated. When the job is executed, the migration data obtained from the source will be replicated, exported, and imported to the destination.

Figure 6-173 Configuring a where clause



The where clause is as follows:

```
dt > '${maxtime}'
```

6.14.7 IF Statements

When developing and orchestrating jobs in DataArts Factory, you can use IF statements to determine the branch to execute.

This section describes how to use IF statements in the following scenarios:

- **Determining the IF Statement Branch to Be Executed Based on the Execution Status of the Previous Node**
- **Determining the IF Statement Branch to Be Executed Based on the Execution Result of the Previous Node**
- **Configuring the Policy for Executing a Node with Multiple IF Statements**

IF statements use EL expressions. You can select EL expressions and follow the instruction in this section to develop jobs.

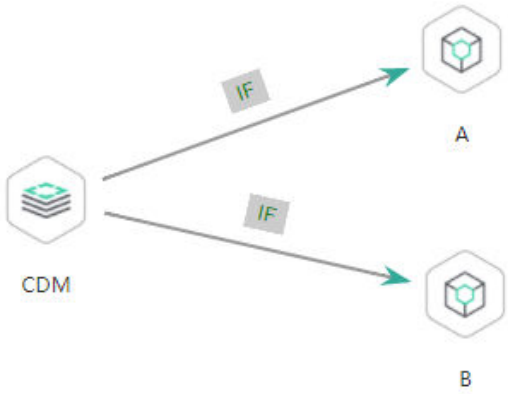
For details about how to use EL expressions, see **EL Expressions**.

Determining the IF Statement Branch to Be Executed Based on the Execution Status of the Previous Node

Scenario

Generally, you can determine the IF statement branch to be executed based on whether the previous CDM node is successfully executed. For details on how to set IF statements, see **Figure 6-174**.

Figure 6-174 Example job



Configuration Method

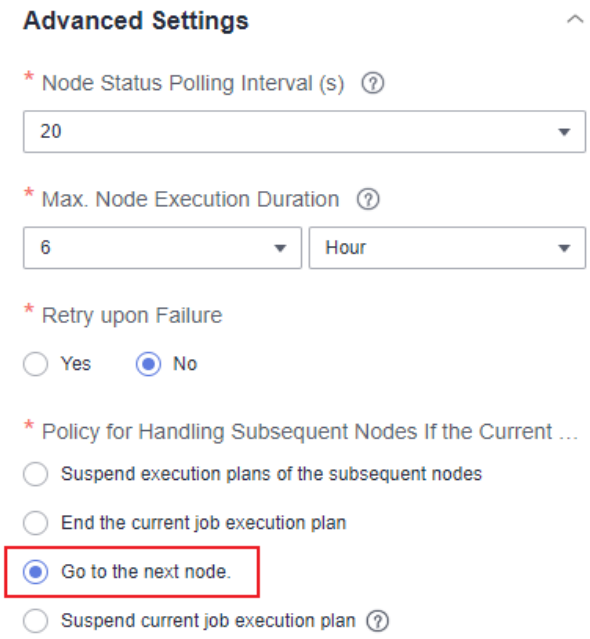
Step 1 Log in to the DataArts Studio console, locate the target DataArts Studio instance, and click **Access** on the instance card.

Step 2 Click the **Workspaces** tab. In the workspace list, locate the target workspace and click **DataArts Factory**. The DataArts Factory console is displayed.

Step 3 On the **Develop Job** page, create a job, drag a CDM node and two Dummy nodes and drop them on the canvas in the right pane. Click and hold to connect the CDM node to the Dummy nodes, as shown in **Figure 6-174**.

Set the **Failure Policy** for the CDM node to **Go to the next node**.

Figure 6-175 Configuring the failure policy for the CDM node



Step 4 Right-click the connection line and select **Set Condition**. In the **Edit EL Expression** dialog box, enter the IF statement in the text box.

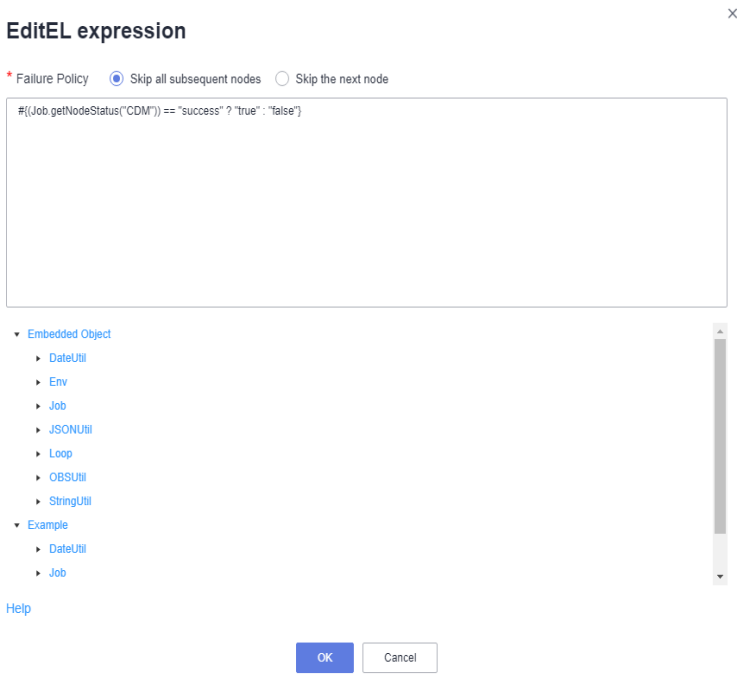
Each statement branch requires an IF statement. The IF statement is a ternary expression based on the EL expression syntax. If the result of the ternary expression is **true**, subsequent nodes will be connected. Otherwise, subsequent nodes will be skipped.

In this demo, the **#{Job.getNodeStatus("node_name")}** EL expression is used to obtain the execution status of a specified node. If the execution is successful, **success** is returned; otherwise, **fail** is returned. In this example, the IF statement expressions are as follows:

- The IF statement expression for branch A is **#{(Job.getNodeStatus("CDM")) == "success" ?**. "true" : "false"}
- The IF statement expression for branch B is **#{(Job.getNodeStatus("CDM")) == "fail" ?**. "true" : "false"}

After entering the IF statement expression, you can select either **Skip all subsequent nodes** or **Skip the next node** for **Failure Policy**. After the configuration is complete, click **OK** to save the job.

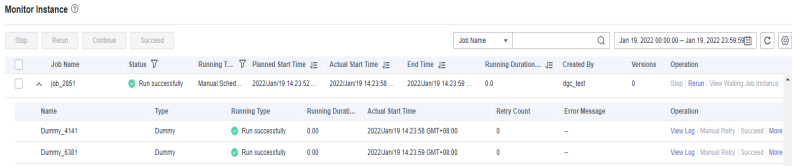
Figure 6-176 Configuring a failure policy



Step 5 Click **Test** to test the job and view the execution result on the **Monitor Instance** page.

Step 6 After the job is executed, view the job instance running result on the **Monitor Instance** page. The execution result meets the expectation. If the execution result is **fail**, branch A is skipped and branch B is executed.

Figure 6-177 Job execution result



----End

Determining the IF Statement Branch to Be Executed Based on the Execution Result of the Previous Node

Scenario Description

Scenario: Use the Hive SQLnode to collect statistics on the number of people whose score is higher than 85, transfer the execution result as a parameter to the next node, compare the result with the number of people who have passed the test, and determine the IF condition branch to be executed.

Analysis: The execution result of the select statement on the Hive SQL node is a two-dimensional array which contains a single field. Therefore, EL expression **#{Loop.dataArray[][]}** or **#{Loop.current[]}** can be used to obtain the value in the two-dimensional array. Currently, only the For Each node supports loop expressions, so the Hive SQL node needs to be connected to a For Each node.

NOTE

In this scenario, the loop expression cannot be replaced by the StringUtil expression **#{StringUtil.split(StringUtil.split(StringUtil.split(Job.getNodeOutput("*Name of the previous node*"),"]")[0],"[")[0],"\\"")[0]}** because the StringUtil expression returns a string which cannot be compared with the standard data of the int type.

Figure 6-178 shows the job orchestration.

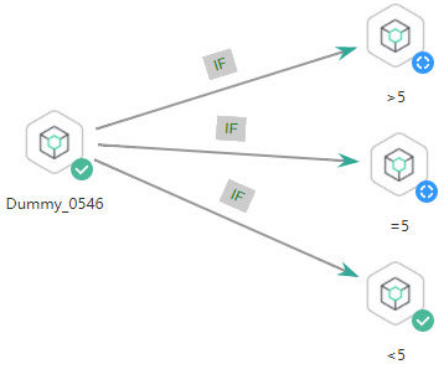
Figure 6-178 Example job

Key configurations of the For Each node are as follows:

- **Dataset**: Enter the execution result of the select statement on the Hive SQL node. Use the **#{Job.getNodeOutput('HIVE')}** expression, where **HIVE** is the name of the previous node.
- **Subjob Parameter Name**: Enter the parameter defined in the subjob. Transfer the output of the previous node of the main job to the sub-job for use. The variable name is **result**, and its value is a column in the dataset. The EL expression **#{Loop.dataArray[0][0]}** or **#{Loop.current[]}** is used. This example uses **{Loop.dataArray[0][0]}** as an example.

The sub-job selected on the For Each node determines the IF statement branch to be executed based on the subjob parameter transferred from the For Each node. **Figure 6-179** shows the job orchestration.

Figure 6-179 Example sub-job



The IF statement is the key configuration of the subjob. This example uses the expression **${result}** to obtain the value of the job parameter.

> **NOTE**
>
> Do not use the **#{Job.getParam("job_param_name")}** EL expression because this expression can only obtain the values of the parameters configured in the current job, but cannot obtain the parameter values transferred from the parent job or the global variables configured in the workspace. The expression only works for the current job.
>
> To obtain the parameter values passed from the parent job and the global variables configured for the workspace, you are advised to use the **${job_param_name}** expression.

Configuration Method

Developing a Subjob

Step 1 Log in to the DataArts Studio console, locate the target DataArts Studio instance, and click **Access** on the instance card.

Step 2 Click the **Workspaces** tab. In the workspace list, locate the target workspace and click **DataArts Factory**. The DataArts Factory console is displayed.

Step 3 On the **Develop Job** page, create a data development subjob For Each. Drag four Dummy nodes and drop them on the canvas, click and hold to connect them, as shown in **Figure 6-179**.

Step 4 Right-click the connection line and select **Set Condition**. In the **Edit EL Expression** dialog box, enter the IF statement in the text box.

Each statement branch requires an IF statement. The IF statement is a ternary expression based on the EL expression syntax. If the result of the ternary expression is **true**, subsequent nodes will be connected. Otherwise, subsequent nodes will be skipped.

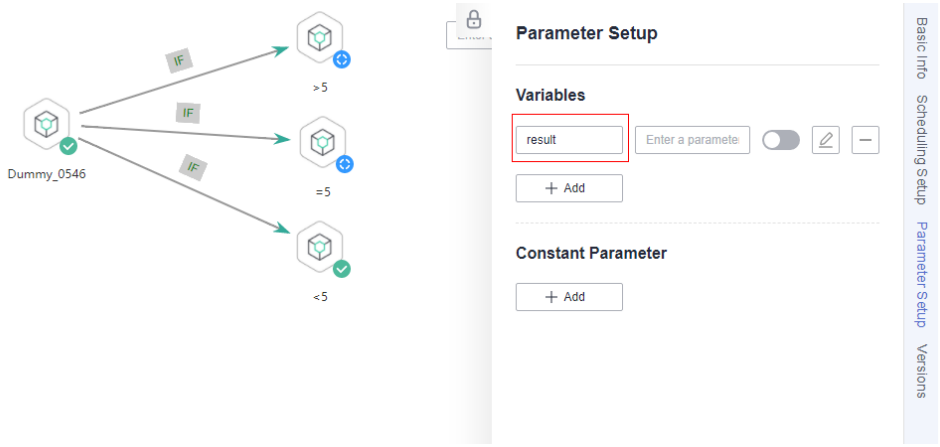
- For the **>5** branch, the IF statement expression is **#{${result} > 5 ? "true" : "false"}**.
- For the **=5** branch, the IF statement expression is **#{${result} == 5 ? "true" : "false"}**.
- For the **<5** branch, the IF statement expression is **#{${result} < 5 ? "true" : "false"}**.

After entering the IF statement expression, you can select either **Skip all subsequent nodes** or **Skip the next node** for **Failure Policy**.

> **NOTE**
>
> If an expression contains multiple conditions, you can use || to combine them conditions. The following is an example:
>
> #{(${result} >= 19 || ${result} <=9) ? "true" : "false"}

Step 5 Configure job parameters. Set the parameter name to **result**. This parameter is only used by the For Each node in the main job **testif** to identify subjob parameters. You do not need to set the parameter value.

Figure 6-180 Configuring job parameters



Step 6 Save the job.

----End

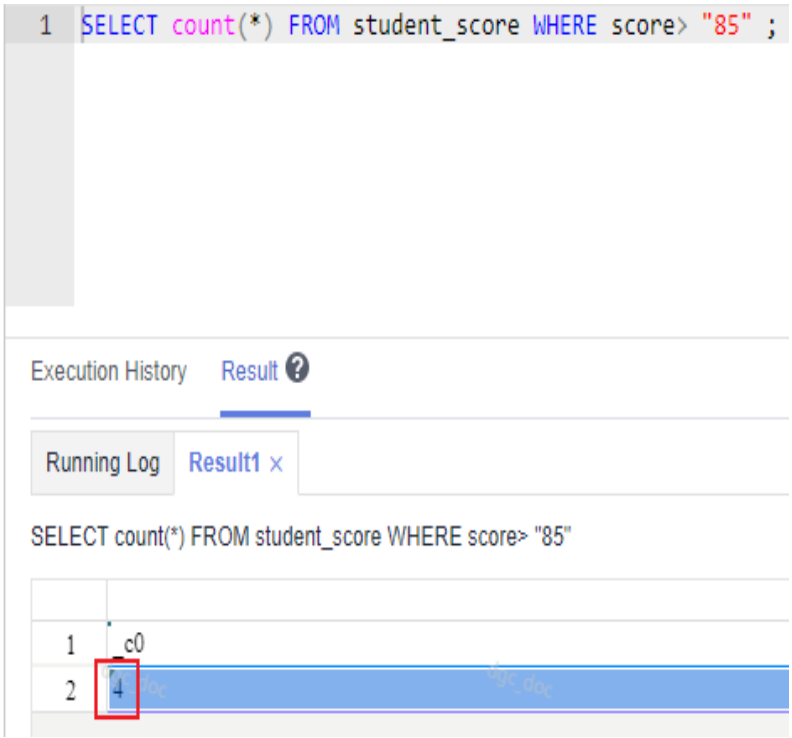
Developing a Job

Step 1 On the **Develop Job** page, create a data development job named **testif**. Drag a HIVE SQL node and a For Each node and drop them on the canvas. Click and hold to connect the nodes, as shown in **Figure 6-178**.

Step 2 Configure properties for the HIVE SQL node. Reference the following SQL script (there is no special requirement for other properties):

```
--Obtain the number of people whose scores are higher than 85 from the student_score table.
SELECT count(*) FROM student_score WHERE score> "85" ;
```

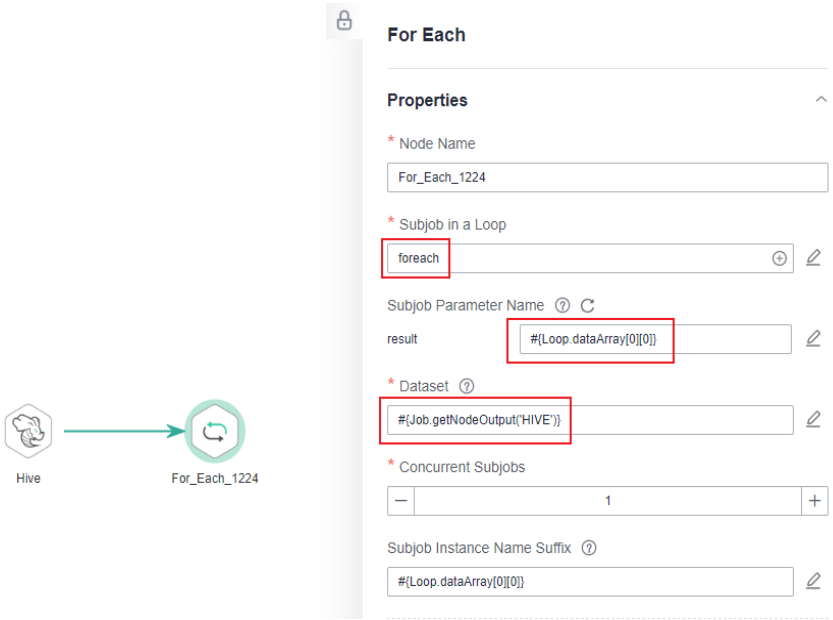
Figure 6-181 HIVE SQL script execution result



Step 3 Configure properties for the For Each node.

- **Subjob in a Loop**: Select **foreach**, the subjob that has been developed.
- **Dataset**: Enter the execution result of the select statement on the Hive SQL node. Use the **#{Job.getNodeOutput('HIVE')}** expression, where **HIVE** is the name of the previous node.
- **Subjob Parameter Name**: Enter the parameter defined in the subjob. Transfer the output of the previous node of the main job to the sub-job for use. The variable name is **result** (parameter name of the subjob), and its value is a column in the dataset. The EL expression **#{Loop.dataArray[0][0]}** is used.

Figure 6-182 Properties of the For Each node



Step 4 Save the job.

----End

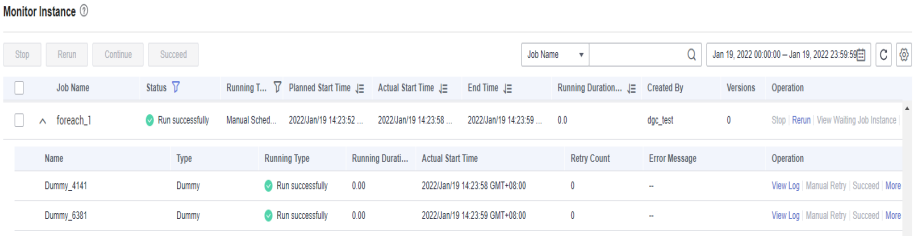
Testing the Main Job

Step 1 Click **Test** above the canvas to test the main job. After the main job is executed, the subjob is automatically invoked through the For Each node and executed.

Step 2 In the navigation pane on the left, choose **Monitor Instance** to view the job execution result.

Step 3 After the job is executed, view the execution result of the subjob **foreach** on the **Monitor Instance** page. The execution result meets the expectation. Currently, the execution result of the Hive SQL statement is **4**. Therefore, the **>5** and **=5** branches are skipped, and the **<5** branch is successfully executed.

Figure 6-183 Execution result of the subjob



----End

Configuring the Policy for Executing a Node with Multiple IF Statements

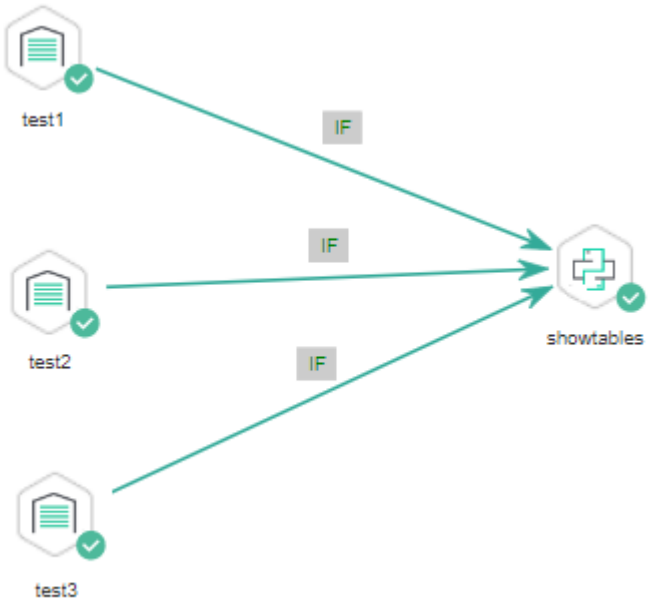
If the execution of a node depends on multiple IF statements, the policy for executing the node can be **AND** or **OR**.

If you choose the **OR** policy, the node will be executed if any one of the IF statements is met.

If you choose the **AND** policy, the node will be executed only if all of the IF statements are met.

If you choose neither, the **OR** policy will be used.

Figure 6-184 A job with multiple IF statements



Configuration Method

Configure the execution policy.

Step 1 Log in to the DataArts Studio console, locate the target DataArts Studio instance, and click **Access** on the instance card.

Step 2 Click the **Workspaces** tab. In the workspace list, locate the target workspace and click **DataArts Factory**. The DataArts Factory console is displayed.

Step 3 On the DataArts Factory console, choose **Configuration** > **Configure** > **Default Configuration**.

Step 4 Select **AND** or **OR** for **Multi-IF Policy**.

Step 5 Click **Save**.

----End

Develop a job.

Step 1 On the **Develop Job** page, create a data development job.

Step 2 Drag three DWS SQL operators as parent nodes and one Python operator as a child node to the canvas. Click and hold to connect the nodes to orchestrate the job shown in **Figure 6-184**.

Step 3 Right-click the connection line and select **Set Condition**. In the **Edit EL Expression** dialog box, enter the IF statement in the text box.

Each statement branch requires an IF statement. The IF statement is a ternary expression based on the EL expression syntax.

- The IF statement expression for the test1 node is **#{(Job.getNodeStatus("test1")) == "success" ? "true" : "false"},**
- The IF statement expression for the test2 node is **#{(Job.getNodeStatus("test2")) == "success" ? "true" : "false"},**
- The IF statement expression for the test3 node is **#{(Job.getNodeStatus("test3")) == "success" ? "true" : "false"},**

The expression of each node is determined using the IF statement based on the execution status of the previous node.

After entering the IF statement expression, you can select either **Skip all subsequent nodes** or **Skip the next node** for **Failure Policy**.

----End

Test the job.

Step 1 Click **Save** above the canvas to save the job.

Step 2 Click **Test** above the canvas to test the job.

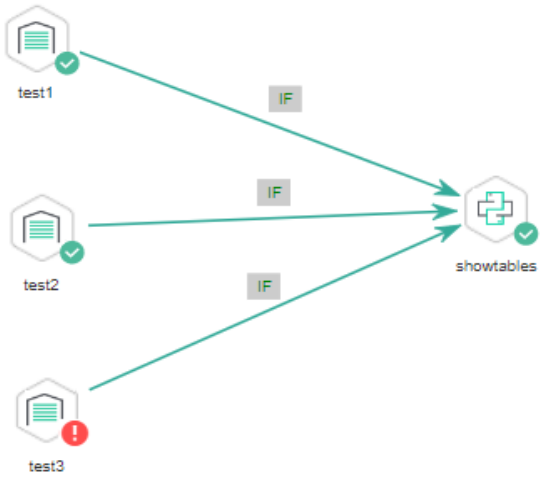
If **test1** is executed successfully, the corresponding IF statement is true.

If **test2** is executed successfully, the corresponding IF statement is true.

If **test3** fails to be executed, the corresponding IF statement is false.

If **Multi-IF Policy** is set to **OR**, the **showtables** node is executed and the job execution is complete.

Figure 6-185 How the job runs if Multi-IF Policy is OR

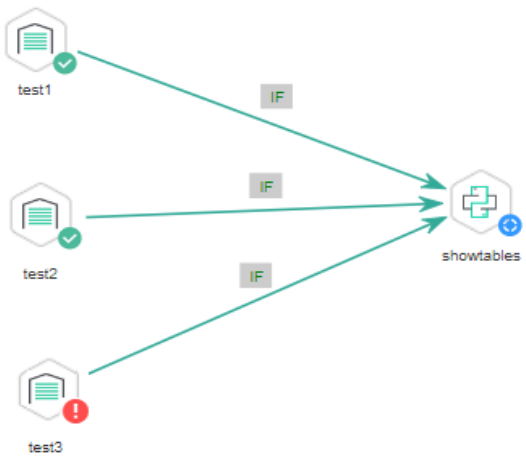


Logs

[INFO][Jul 04, 2022 17:28:23 GMT+08:00] : The job starts to run.
[INFO][Jul 04, 2022 17:30:31 GMT+08:00] : Node test1 started to run.
[INFO][Jul 04, 2022 17:30:31 GMT+08:00] : Node test2 started to run.
[INFO][Jul 04, 2022 17:30:31 GMT+08:00] : Node test3 started to run.
[ERROR][Jul 04, 2022 17:30:51 GMT+08:00] : Node test3 failed to run.
[INFO][Jul 04, 2022 17:30:51 GMT+08:00] : Node test1 finished to run.
[INFO][Jul 04, 2022 17:30:51 GMT+08:00] : Node test2 finished to run.
[INFO][Jul 04, 2022 17:30:51 GMT+08:00] : Node showtables started to run.
[INFO][Jul 04, 2022 17:30:51 GMT+08:00] : Node showtables finished to run.
[INFO][Jul 04, 2022 17:30:51 GMT+08:00] : Job running is completed.

If **Multi-IF Policy** is set to **AND**, the **showtables** node is skipped and the job execution is complete.

Figure 6-186 How the job runs if Multi-IF Policy is AND



Logs

[INFO][Jul 05, 2022 09:05:33 GMT+08:00] : The job starts to run.
[INFO][Jul 05, 2022 09:07:42 GMT+08:00] : Node test1 started to run.
[INFO][Jul 05, 2022 09:07:42 GMT+08:00] : Node test2 started to run.
[INFO][Jul 05, 2022 09:07:42 GMT+08:00] : Node test3 started to run.
[ERROR][Jul 05, 2022 09:08:03 GMT+08:00] : Node test3 failed to run.
[INFO][Jul 05, 2022 09:08:03 GMT+08:00] : Node test1 finished to run.
[INFO][Jul 05, 2022 09:08:03 GMT+08:00] : Node test2 finished to run.
[INFO][Jul 05, 2022 09:08:03 GMT+08:00] : Node showtables finished to run.

----End

6.14.8 Obtaining the Return Value of a Rest Client Node

The Rest Client node can execute RESTful requests on Huawei Cloud.

This tutorial describes how to obtain the return value of the Rest Client node, covering the following two application scenarios:

- **Obtaining the Return Value Through Parameter "The response message body parses the transfer parameter"**
- **Obtaining the Return Value Using an EL Expression**

Obtaining the Return Value Through Parameter "The response message body parses the transfer parameter"

As shown in **Figure 6-187**, the first Rest Client node invokes the API of MRS to query the cluster list. **Figure 6-188** shows the JSON message body returned by the API.

- Scenario: The ID of the first cluster in the cluster list needs to be obtained and transferred to other nodes as a parameter.

- Key configurations: Set **The response message body parses the transfer parameter** of the first Rest Client to **clusterId=clusters[0].clusterId**. Other Rest Client nodes can reference the ID of the first cluster in ${clusterId} mode.

  NOTE

  When setting **The response message body parses the transfer parameter**, ensure that the transferred parameter name (for example, **clusterId**) is unique among all node parameters of the job.

Figure 6-187 Rest Client job example 1

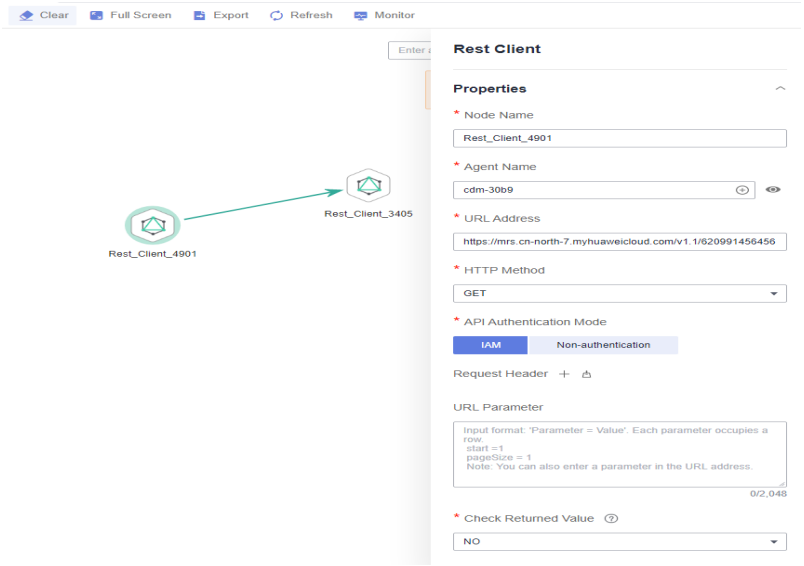


Figure 6-188 JSON message body

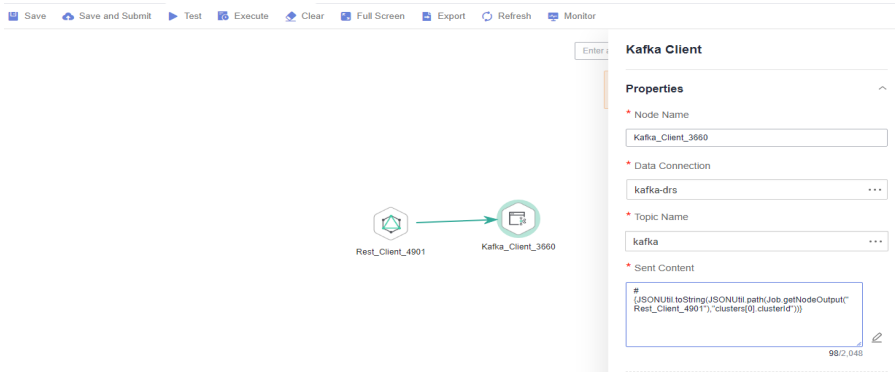
Obtaining the Return Value Using an EL Expression

The Rest Client node can be used together with EL expressions. You can select different EL expressions based on scenarios. This section describes how to develop your own jobs based on your service requirements. For details about how to use EL expressions, see **EL Expressions**.

As shown in **Figure 6-189**, the Rest Client invokes the API of MRS to query the cluster list and then invokes the Kafka Client to send a message.

- Scenario: The Kafka Client sends a character string message. The message content is the ID of the first cluster in the cluster list.
- Key configurations: When you configure the Kafka Client, use the following EL expression to obtain a specific field in the message body returned by the REST API:
  #{JSONUtil.toString(JSONUtil.path(Job.getNodeOutput("Rest_Client_4901"),"clusters[0].clusterId"))}

Figure 6-189 Rest Client job example 2



6.14.9 Using For Each Nodes

Scenario

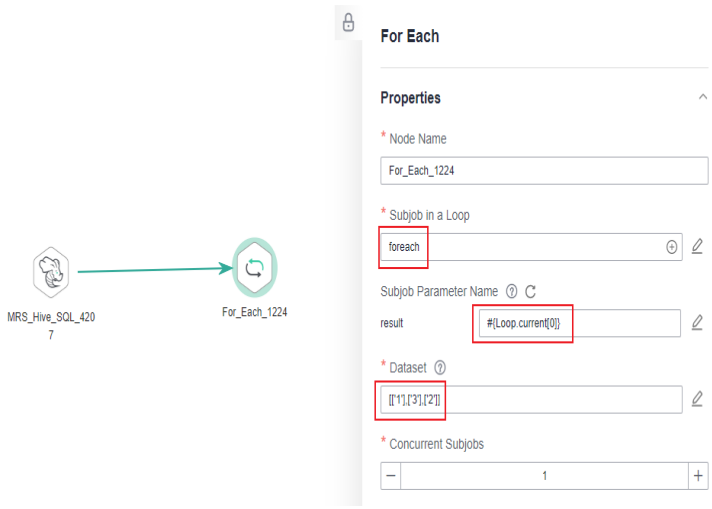
During job development, if some jobs have different parameters but the same processing logic, you can use For Each nodes to avoid repeated job development.

You can use a For Each node to execute a subjob in a loop and use a dataset to replace the parameters in the subjob. The key parameters are as follows:

- **Subjob in a Loop**: Select the subjob to be executed in a loop.
- **Dataset**: Enter a set of parameter values of the subjobs. The value can be a specified dataset such as **[['1'],['3'],['2']]** or an EL expression such as **#{Job.getNodeOutput('preNodeName')}**, which is the output value of the previous node.
- **Subjob Parameter Name**: The parameter name is the variable defined in the subjob. The parameter value is usually set to a group of data in the dataset. Each time the job is run, the parameter value is transferred to the subjob for use. For example, parameter value **#{Loop.current[0]}** indicates that the first value of each row of data in the dataset is traversed and transferred to the subjob.

Figure 6-190 shows an example For Each node. As shown in the figure, the parameter name of the **foreach** subjob is **result**, and the parameter value is the traversal of the one-dimensional array dataset **[['1'],['3'],['2']]** (that is, the value is **1**, **3**, and **2** in the first, second, and third loop, respectively).

Figure 6-190 For Each node



For Each Nodes and EL Expressions

To use For Each nodes properly, you must be familiar with EL expressions. For details about how to use EL expressions, see **EL Expressions**.

For Each nodes use the following EL expressions most:

- #{Loop.dataArray}: dataset input by the For Each node. It is a two-dimensional array.
- #{Loop.current}: The For Loop node processes a dataset line by line. *Loop.current* indicates a line of data that is being processed. *Loop.current* is a one-dimensional array, and its format is #{Loop.current[0]}, #{Loop.current[1]}, or others. The value 0 indicates that the first value in the current line is traversed.
- #{Loop.offset}: current offset when the For Each node processes the dataset. The value starts from 0.
- #{Job.getNodeOutput('preNodeName')}: obtains the output of the previous node.

Examples

Scenario

To meet data normalization requirements, you need to periodically import data from multiple source DLI tables to the corresponding destination DLI tables, as listed in **Table 1**.

Table 6-186 Tables to be imported

Source Table	Destination Table
a_new	a
b_2	b
c_3	c
d_1	d
c_5	e
b_1	f

If you use SQL nodes to execute import scripts, a large number of scripts and nodes need to be developed, resulting in repeated work. In this case, you can use the For Each node to perform cyclic jobs to reduce the development workload.

Configuration Method

Step 1 Prepare the source and destination tables. To facilitate subsequent job execution and verification, you need to create a source DLI table and a destination DLI table and insert data into the tables.

1. Create a DLI table. You can create a DLI SQL script on the **DataArts Factory** page and run the following commands to create a DLI table. You can also run the following SQL commands in the SQL editor on the DLI console.

```
/* Create a data table. */
CREATE TABLE a_new (name STRING, score INT) STORED AS PARQUET;
CREATE TABLE b_2 (name STRING, score INT) STORED AS PARQUET;
CREATE TABLE c_3 (name STRING, score INT) STORED AS PARQUET;
CREATE TABLE d_1 (name STRING, score INT) STORED AS PARQUET;
CREATE TABLE c_5 (name STRING, score INT) STORED AS PARQUET;
CREATE TABLE b_1 (name STRING, score INT) STORED AS PARQUET;
CREATE TABLE a (name STRING, score INT) STORED AS PARQUET;
CREATE TABLE b (name STRING, score INT) STORED AS PARQUET;
CREATE TABLE c (name STRING, score INT) STORED AS PARQUET;
CREATE TABLE d (name STRING, score INT) STORED AS PARQUET;
CREATE TABLE e (name STRING, score INT) STORED AS PARQUET;
CREATE TABLE f (name STRING, score INT) STORED AS PARQUET;
```

2. Insert data into the source data table. You can create a DLI SQL script on the **DataArts Factory** page and run the following commands to create a DLI table. You can also run the following SQL commands in the SQL editor on the DLI console.

```
/* Insert data into the source data table. */
INSERT INTO a_new VALUES ('ZHAO','90'),('QIAN','88'),('SUN','93');
INSERT INTO b_2 VALUES ('LI','94'),('ZHOU','85');
INSERT INTO c_3 VALUES ('WU','79');
INSERT INTO d_1 VALUES ('ZHENG','87'),('WANG','97');
INSERT INTO c_5 VALUES ('FENG','83');
INSERT INTO b_1 VALUES ('CEHN','99');
```

Step 2 Prepare dataset data. You can obtain a dataset in any of the following ways:

1. Import the data in **Table 1** into the DLI table and use the result read by the SQL script as the dataset.
2. You can save the data in **Table 1** to a CSV file in the OBS bucket. Then use a DLI SQL or DWS SQL statement to create an OBS foreign table, associate it

with the CSV file, and use the query result of the OBS foreign table as the dataset. For details about how to create a foreign table on DLI, see **OBS Source Stream**. For details about how to create a foreign table on DWS, see **Creating a Foreign Table**.

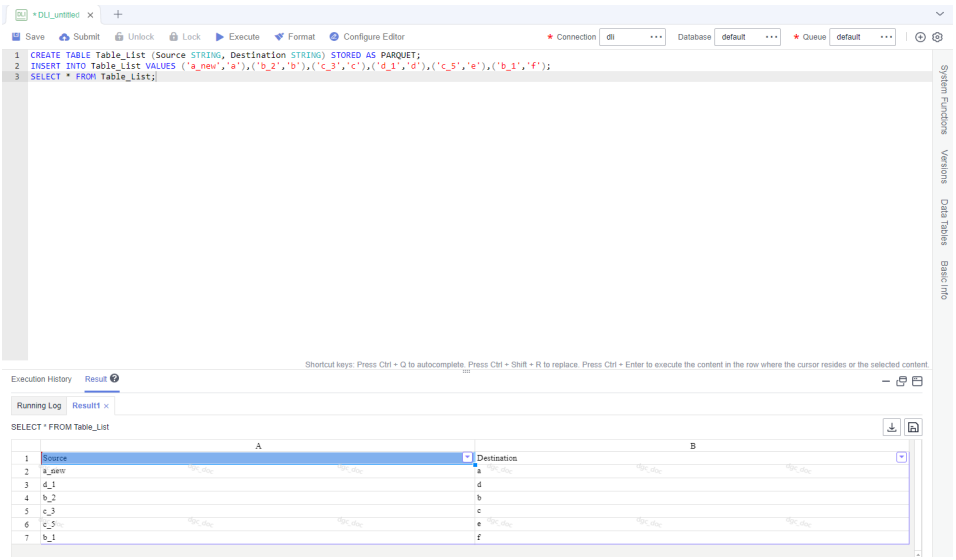
3. You can save the data in **Table 1** to a CSV file in the HDFS. Then use a Hive SQL statement to create a Hive foreign table, associate it with the CSV file, and use the query result of the Hive foreign table as the dataset. For details about how to create an MRS foreign table, see **Creating a Table**.

This section uses method 1 as an example to describe how to import data from **Table 1** to the DLI table (**Table_List**). You can create a DLI SQL script on the **DataArts Factory** page and run the following commands to import data into the table. You can also run the following SQL commands in the SQL editor on the DLI console.

```
/* Create the Table_List data table, insert data in Table 1 into the table, and check the generated data. */
CREATE TABLE Table_List (Source STRING, Destination STRING) STORED AS PARQUET;
INSERT INTO Table_List VALUES ('a_new','a'),('b_2','b'),('c_3','c'),('d_1','d'),('c_5','e'),('b_1','f');
SELECT * FROM Table_List;
```

The generated data in the **Table_List** table is as follows:

Figure 6-191 Data in the Table_List table



Step 3 Create a subjob named **ForeachDemo** to be executed cyclically. In this operation, a task containing the DLI SQL node is defined to be executed cyclically.

1. Access the DataArts Studio **DataArts Factory** page, choose **Develop Job**. Create a job named **ForeachDemo**, select the DLI SQL node, and configure the job as shown in **Figure 6-192**.

   In the DLI SQL statement, set the variable to be replaced to **${}**. The following SQL statement is used to import all data in the **${Source}** table to the **${Destination}** table. **${fromTable}** and **${toTable}** are the variables. The SQL statement is as follows:

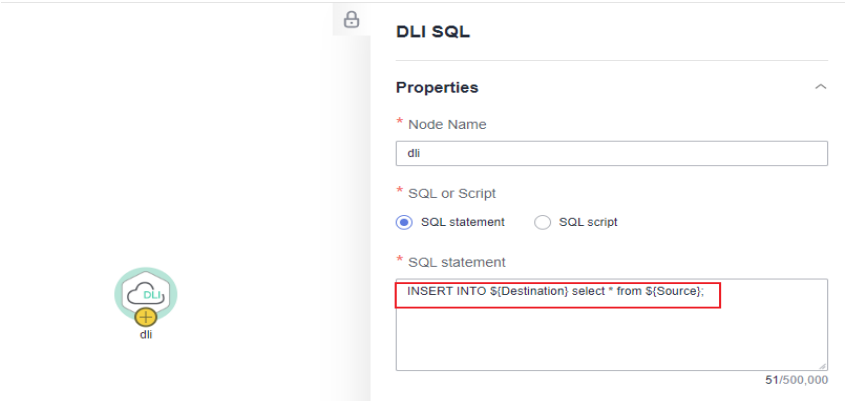
   ```
   INSERT INTO ${Destination} select * from ${Source};
   ```


NOTE

Do not use the **#{Job.getParam("job_param_name")}** EL expression because this expression can only obtain the values of the parameters configured in the current job, but cannot obtain the parameter values transferred from the parent job or the global variables configured in the workspace. The expression only works for the current job.

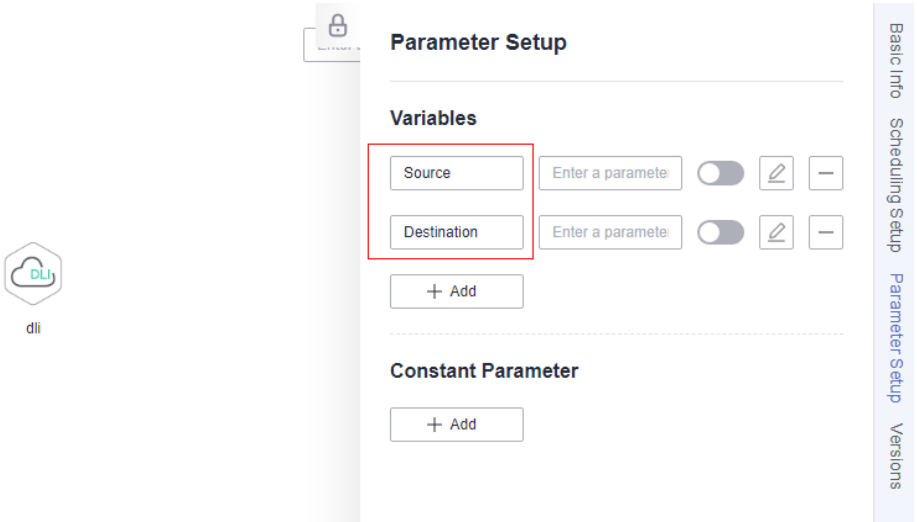
To obtain the parameter values passed from the parent job and the global variables configured for the workspace, you are advised to use the **${job_param_name}** expression.

Figure 6-192 Cyclically executing a subjob



2. After configuring the SQL statement, configure parameters for the subjob. You only need to set the parameter names, which are used by the For Each operator of the **ForeachDemo_master** job to identify subjob parameters.

Figure 6-193 Configuring subjob parameters



3. Save the job.

Step 4 Create a master job named **ForeachDemo_master** where the For Each node is located.

1. Access the DataArts Studio **DataArts Studio** page and choose **Develop Job**. Create a data development master job named **ForeachDemo_master**. Select

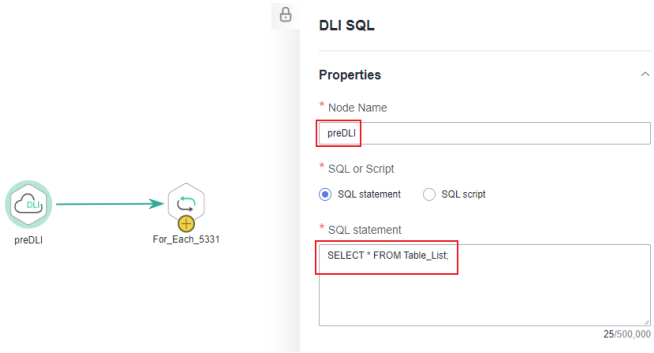
the DLI SQL and For Each nodes and click and drag to compile the job shown in **Figure 6-194**.

Figure 6-194 Compiling a job

2. Configure the properties of the DLI SQL node. Select **SQL statement** and and enter the following statement. The DLI SQL node reads data from the DLI table **Table_List** and uses it as the dataset.

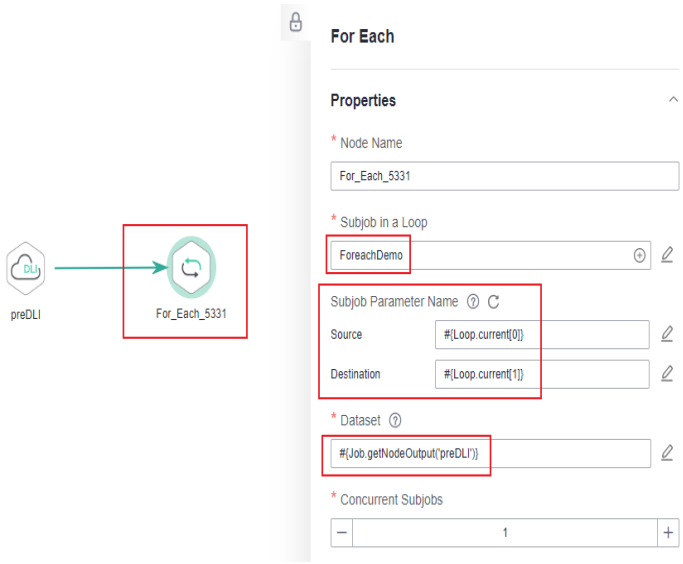
```
SELECT * FROM Table_List;
```

Figure 6-195 DLI SQL node configuration



3. Configure properties for the For Each node.
   - **Subjob in a Loop**: Select **ForeachDemo**, which is the subjob that has been developed in **step 2**.
   - **Dataset**: Enter the execution result of the select statement on the DLI SQL node. Use the **#{Job.getNodeOutput('preDLI')}** expression, where **preDLI** is the name of the previous node.
   - **Subjob Parameter Name**: used to transfer data in the dataset to the subjob **Source** corresponds to the first column in the **Table_List** table of the dataset, and **Destination** corresponds to the second column. Therefore, enter EL expression **#{Loop.current[0]}** for **Source** and **#{Loop.current[1]}** for **Destination**.

Figure 6-196 Configuring properties for the For Each node

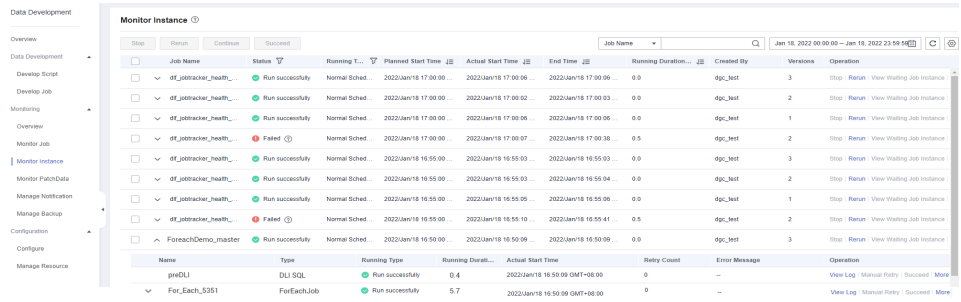


4. Save the job.

Step 5 Test the main job.

1. Click **Test** above the canvas to test the main job. After the main job is executed, the subjob is automatically invoked through the For Each node and executed.
2. In the navigation pane on the left, choose **Monitor Instance** to view the job execution status. After the job is successfully executed, you can view the subjob instances generated on the For Each node. Because the dataset contains six rows of data, six subjob instances are generated.

Figure 6-197 Viewing job instances

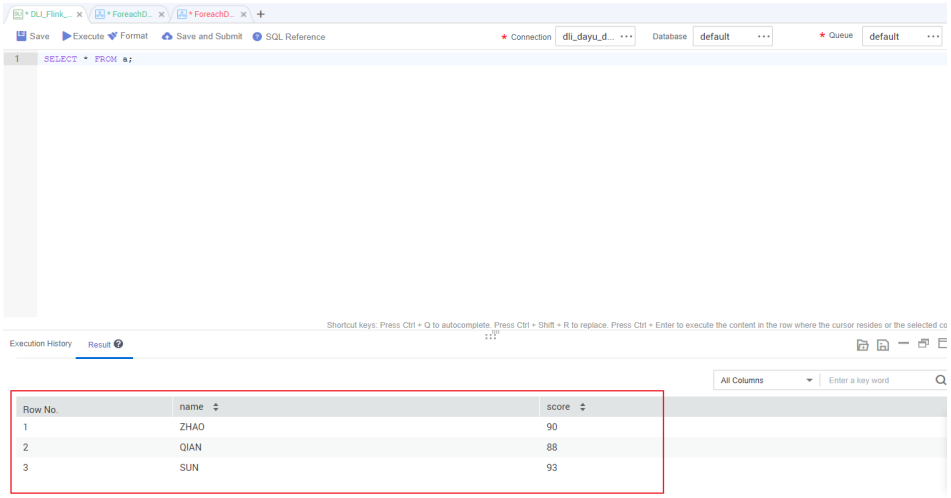


3. Check whether the data has been inserted into the six DLI destination tables. You can create a DLI SQL script on the **DataArts Factory** page and run the following commands to import data into the table. You can also run the following SQL commands in the SQL editor on the DLI console.

```
/* Run the following command to query the data in a table (table a is used as an example): */
SELECT * FROM a;
```

Compare the obtained data with the data in **Insert data into the source data table**. The inserted data meets the expectation.

Figure 6-198 Destination table data



----End

More Cases for Reference

For Each nodes can work with other nodes to implement more functions. You can refer to the following cases to learn more about how to use For Each nodes.

- **Creating Table Migration Jobs in Batches Using CDM Nodes**
- **Determining the IF Statement Branch to Be Executed Based on the Execution Result of the Previous Node**

6.14.10 Using Script Templates and Parameter Templates

Scenario

This function applies to the following scenarios:

- Use a script template for a Flink SQL script.
- During pipeline job development, use a Fink SQL script which uses a script template for the MRS Flink Job node and use a parameter template for **Program Parameter** of the MRS Flink Job node.
- Use a script template in a single-task Flink SQL job.
- Use template parameters in a single-task Flink JAR job.

NOTE

When you use a script template in a script, ensure that the SQL statement is in **@@{Script template}** format.

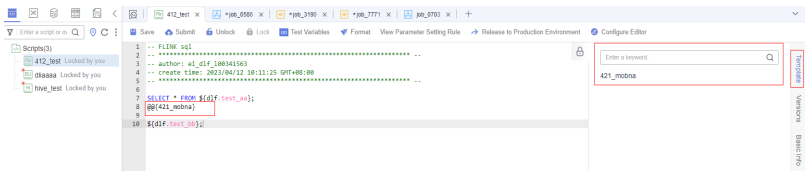
Prerequisites

A template has been created. If no template is available, create one by referring to **Configuring a Template**.

Using Templates

- Use a script template for a Flink SQL script.
  a. In the navigation pane on the DataArts Studio console, choose **Data Development** > **Develop Script**.
  b. Right-click a script directory and select **Create Flink SQL Script**.
  c. Click **Template**. In the slide-out pane, select a template, for example, **412_mobna**. You can select multiple templates.

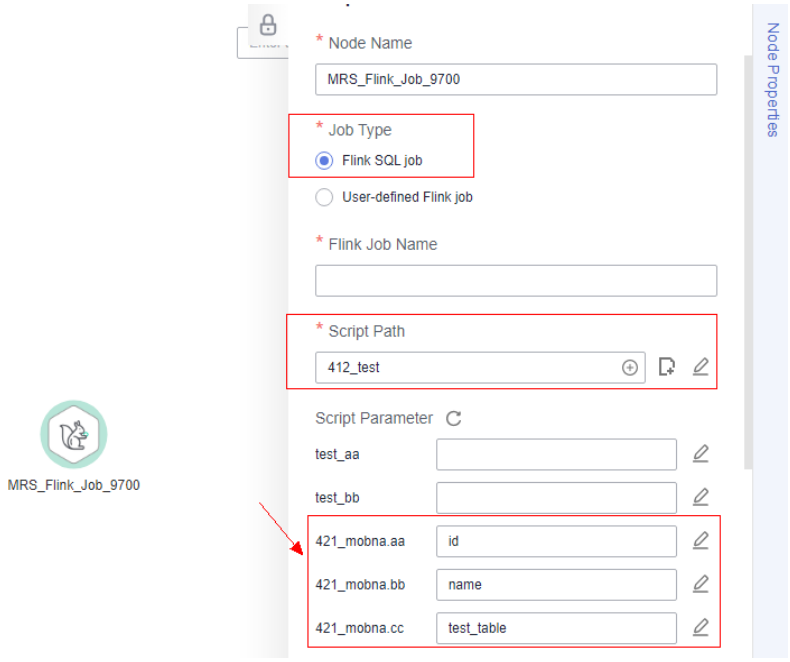
     Figure 6-199 Using a script template

  d. Click **Save** to create the **412_test** script.
- During the development of a pipeline job, use the Flink SQL script which uses a script template for the MRS Flink Job node.
  a. In the navigation pane on the DataArts Factory console, choose **Data Development** > **Develop Job**.
  b. Right-click a job directory and select **Create Job** to create a batch processing job in pipeline mode.
  c. On the displayed data development page, drag an MRS Flink Job node to the canvas.
  d. Select **Flink SQL job** for **Job Type** and select the Flink SQL script for **Script Path**.

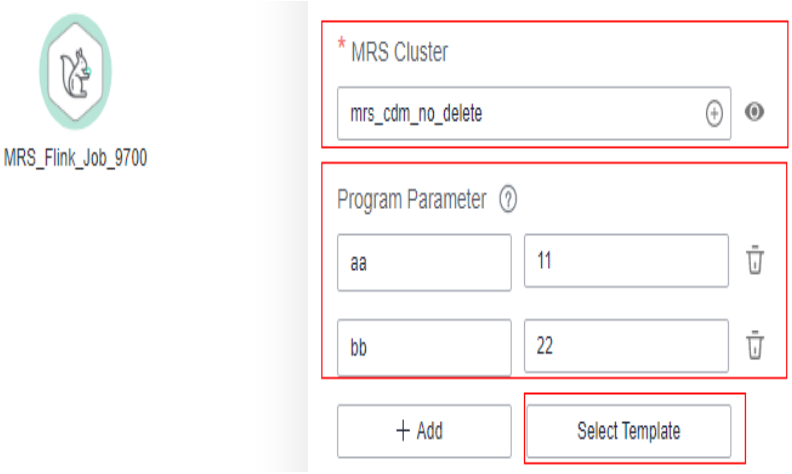
     After the script is selected, the template parameters and values used by the script are automatically displayed.

Figure 6-200 Using the Flink SQL script



- During the development of a pipeline job, use a parameter template in **Program Parameter** of the MRS Flink Job node.
  a. Set **MRS Cluster**.
  b. Program parameters are automatically displayed. Click **Select Template** and select a parameter template. You can also select multiple templates.
  The parameter names and values are automatically displayed.

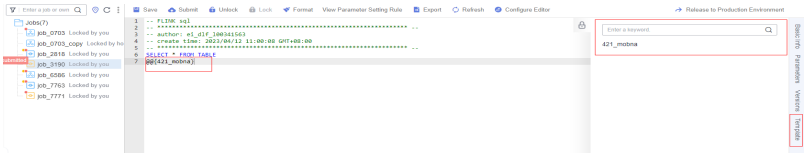
  Figure 6-201 Using a parameter template for program parameters



- Use a script template in a single-task Flink SQL job.
  a. In the navigation pane on the DataArts Factory console, choose **Data Development** > **Develop Job**.
  b. Right-click a job directory and select **Create Job** to create a real-time processing job in single-task Flink SQL mode.

c. Click **Template**. In the slide-out pane, select a template, for example, **412_mobna**. You can select multiple templates.

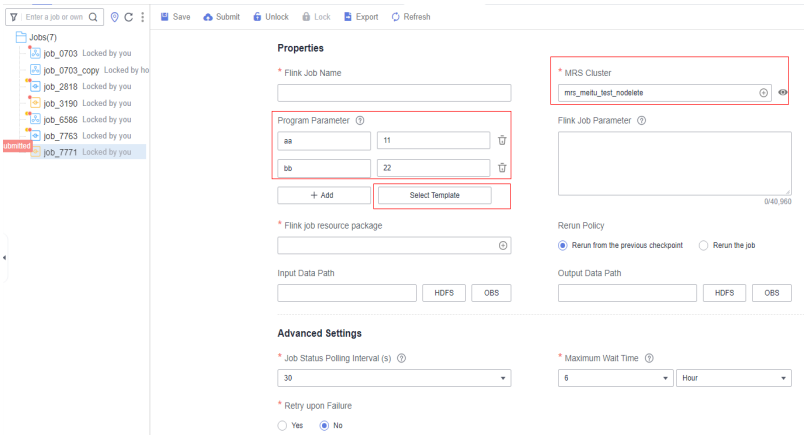
Figure 6-202 Using a script template in a single-task Flink SQL job.



- Use template parameters in a single-task Flink JAR job.
  a. In the navigation pane on the DataArts Factory console, choose **Data Development** > **Develop Job**.
  b. Right-click a job directory and select **Create Job** to create a real-time processing job in single-task Flink JAR mode.
  c. Set **MRS Cluster**.
  d. Program parameters are automatically displayed. Click **Select Template** and select a parameter template. You can also select multiple templates.

  The parameter names and values are automatically displayed.

Figure 6-203 Using a script template in a single-task Flink JAR job.



6.14.11 Developing a Python Job

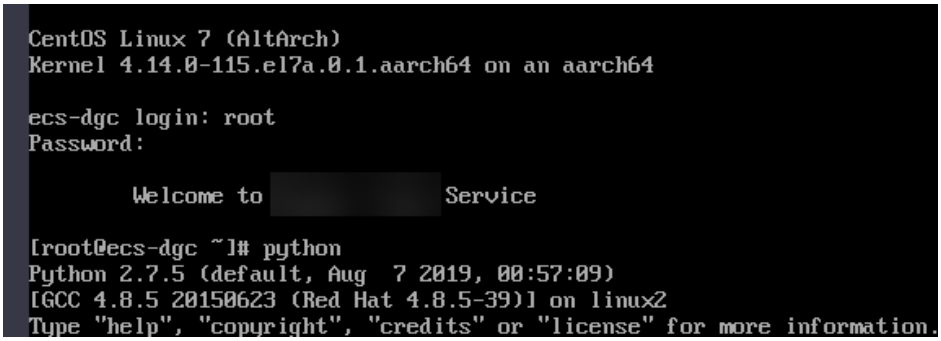
This section describes how to develop and execute a Python job using DataArts Factory.

Preparing the Environment

- An ECS named **ecs-dgc** has been created.

NOTE

In this example, the ECS uses the **CentOS 8.0 64bit with ARM (40 GB)** public image and the Python environment. You can log in to the ECS and run the **python** command to check the Python environment.



- You have enabled the DataArts Migration incremental package and created a CDM cluster named **cdm-dlfpyhthon**. The cluster provides an agent for the DataArts Factory module to communicate with the ECS.
- Ensure that the ECS can communicate with the CDM cluster, which depends on the following conditions:
  - If the CDM cluster and the ECS are in the same region, VPC, subnet, and security group, they can communicate with each other by default. If they are in the same VPC but in different subnets or security groups, you must configure routing rules and security group rules. For details about how to configure routing rules, see **configuring routes**. For details about how to configure security group rules, see **configuring security group rules**.
  - If the CDM cluster and the ECS are in different regions, a public network or a dedicated connection is required for enabling communication between the CDM cluster and the cloud service. If the Internet is used for communication, ensure that an EIP has been bound to the CDM cluster, the host where the data source is located can access the Internet, and the port has been enabled in the firewall rules.
  - The ECS and the CDM cluster belong to the same enterprise project. If they do not, you can modify the enterprise project of the workspace.

Constraints

- Python nodes support script parameters and job parameters.
- This section uses Python3 as an example.

Creating an ECS Data Connection

Before developing a Python script, you need to create a connection to the ECS.

Step 1 Log in to the DataArts Studio console by following the instructions in **Accessing the DataArts Studio Instance Console**.

Step 2 On the DataArts Studio console, locate a workspace and click **Management Center**.

Step 3 On the displayed **Manage Data Connections** page, click **Create Data Connection**.

Step 4 Configure parameters by referring to **Table 6-187** and create a data connection named **ecs**.

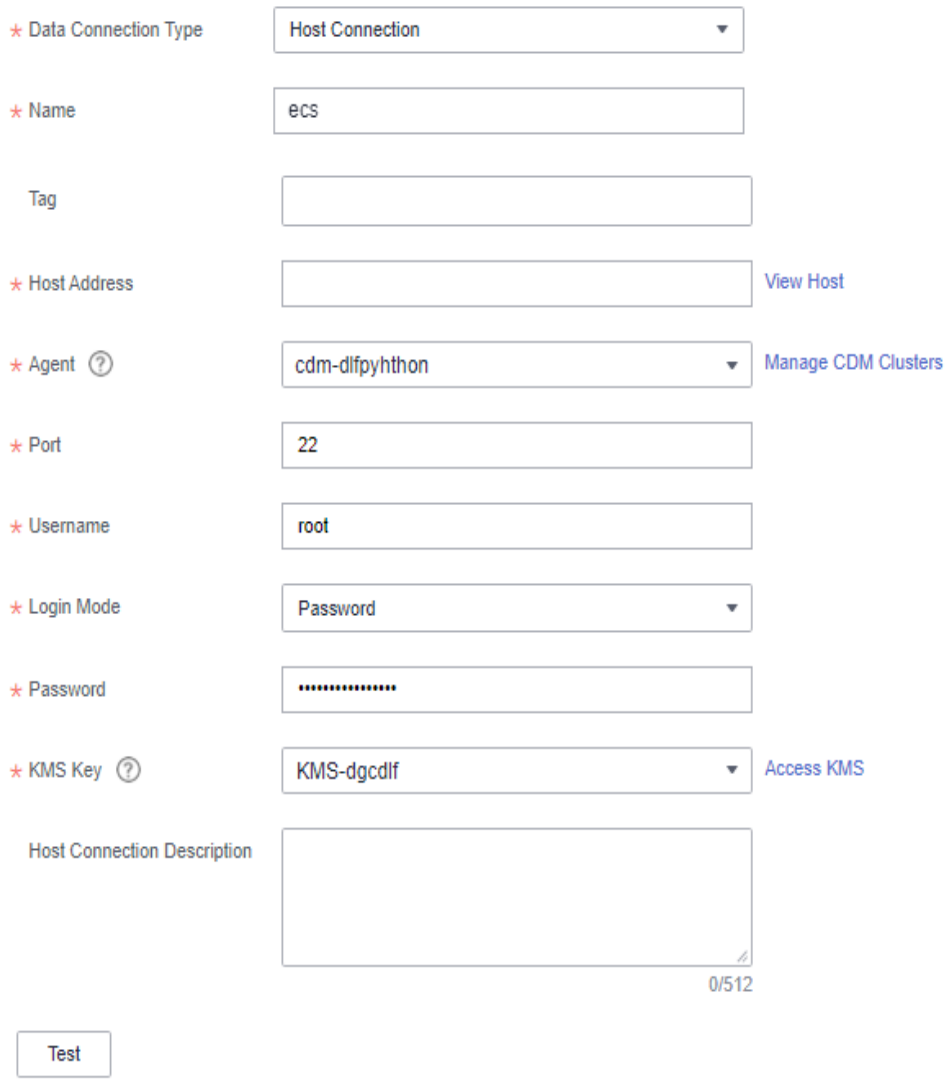
Table 6-187 Host Connection parameters

Parameter	Mandatory	Description
Data Connection Type	Yes	**Host Connection** is selected by default and cannot be changed.
Name	Yes	Name of the data connection to create. Data connection names can contain a maximum of 100 characters. They can contain only letters, digits, underscores (_), and hyphens (-).
Tag	No	Attribute of the data connection to create. Tags make management easier. **NOTE** The tag name can contain only letters, digits, and underscores (_) and cannot start with an underscore (_) or contain more than 100 characters.
Applicable Modules	Yes	Select the modules for which this connection is available. All modules are selected by default, which means this connection is available for all the modules that support the data source connected by this connection. For details about the data sources supported by each module, see **Data Sources**.
Basic and Network Connectivity Configuration		
Host Address	Yes	IP address of the Linux host For details, see **Viewing Details About an ECS**.

Parameter	Mandatory	Description
Agent	Yes	CDM cluster used as an agent. **NOTE** • If a CDM cluster functions as the agent for a data connection in Management Center, the cluster supports a maximum of 200 concurrent active threads. If multiple data connections share an agent, a maximum of 200 SQL, Shell, and Python scripts submitted through the connections can run concurrently. Excess tasks will be queued. You are advised to plan multiple agents based on the workload. • When scheduling shell or Python scripts, the agent accesses the ECS. If shell and Python scripts are scheduled frequently, the ECS adds the private IP address of the agent to the blocklist. To ensure normal job scheduling, you are advised to use the **root** user of the ECS to add the private IP address bound to the agent (CDM cluster) to the **/etc/hosts.allow** file. For details about how to obtain the private IP address of the CDM cluster, see **Viewing Basic Cluster Information and Modifying Cluster Configurations**.
Port	Yes	SSH port number of the host. By default, port 22 is used to log in to a Linux host. If the port number has been changed, you can obtain the new port number from the **port** field in the **/etc/ssh/sshd_config** file.
KMS Key	Yes	KMS key used to encrypt and decrypt the authentication information for the data source
Data Source Authentication and Other Function Configuration		
Username	Yes	Username for logging in to the host
Login Mode	Yes	Mode for logging in to the host • Key Pair • Password
Key Pair	Yes	This parameter is available only when **Login Mode** is set to **Key Pair**. If **Key Pair** is the login mode of the host, you need to obtain the private key file, upload it to OBS, and select an OBS path. **NOTE** The uploaded private key must match the public key configured on the host. For details, see **Application Scenarios for Using Key Pairs**.
Key Pair Password	Yes	If no password is set for the key pair, you do not need to set this parameter.

Parameter	Mandatory	Description
Password	Yes	This parameter is available only when **Login Mode** is set to **Password**. If the login mode of the host is to use a password, enter a login password.
Host Connection Description	No	Descriptive information about the host connection

Figure 6-204 Creating a host connection



NOTE

The key parameters are as follows:

- **Host Address**: Enter the IP address of the **ECS**.
- **Agent**: Select the CDM cluster you have obtained from the **DataArts Migration incremental package**.

Step 5 Click **Test** to test connectivity of the data connection. If the test fails, the data connection cannot be created.

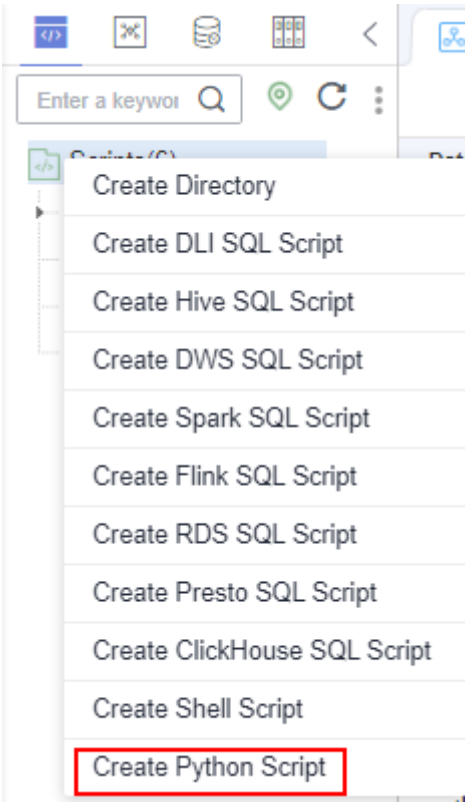
Step 6 After the test is successful, click **OK** to create the data connection.

----End

Developing a Python Script

Step 1 Choose **DataArts Factory** > **Develop Script** and create a Python script named **python_test**.

Figure 6-205 Creating a Python script



Step 2 Select a Python version and host connection, and set input parameters as needed.

NOTE

The parameters will be transferred to the Python script when the script is executed. The parameters are separated by spaces, for example, **Microsoft Oracle**. The parameters must be referenced by the Python script. Otherwise, the parameters are invalid.

Step 3 Edit Python statements in the editor.

This example defines a string template for saving company information and uses the template to output information about different companies.

```
import sys
Company_Name1=sys.argv[1]
Company_Name2=sys.argv[2]
template='No.:{:0>9s} \t CompanyName: {:s} \t Website: https://www.{:s}.com'
context1=template.format('1',Company_Name1,Company_Name1.lower())
context2=template.format('2',Company_Name2,Company_Name2.lower())
print(context1)
print(context2)
```

NOTE

- The script development area in **Figure 6-206** is a temporary debugging area. After you close the script tab, the development area will be cleared.
- **Connection**: Select the data connection created in **Creating an ECS Data Connection**.

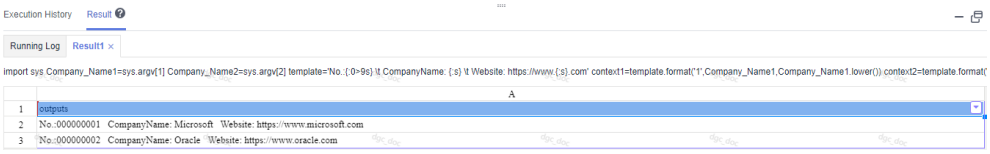
Figure 6-206 Editing the Python statements

Step 4 Click **Save** and then **Submit**.

Step 5 Click **Execute** to execute the Python statements.

Step 6 View the script execution result.

Figure 6-207 Viewing the script execution result



----End

Referencing the Python Script in a Job

Step 1 Create a job.

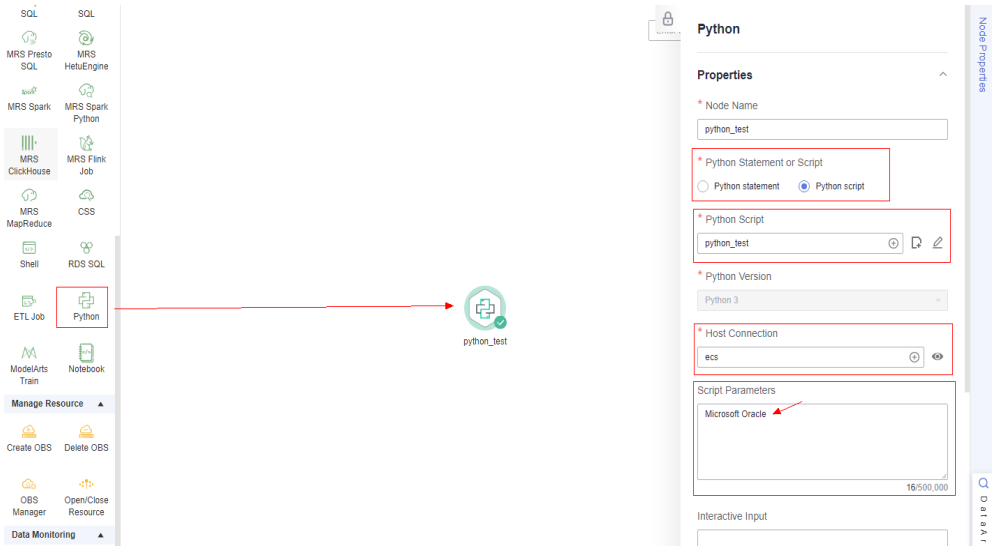
Step 2 Select a Python node and configure the node properties.

Select the created Python script and set the node parameters. Set **Script Parameters**.

NOTE

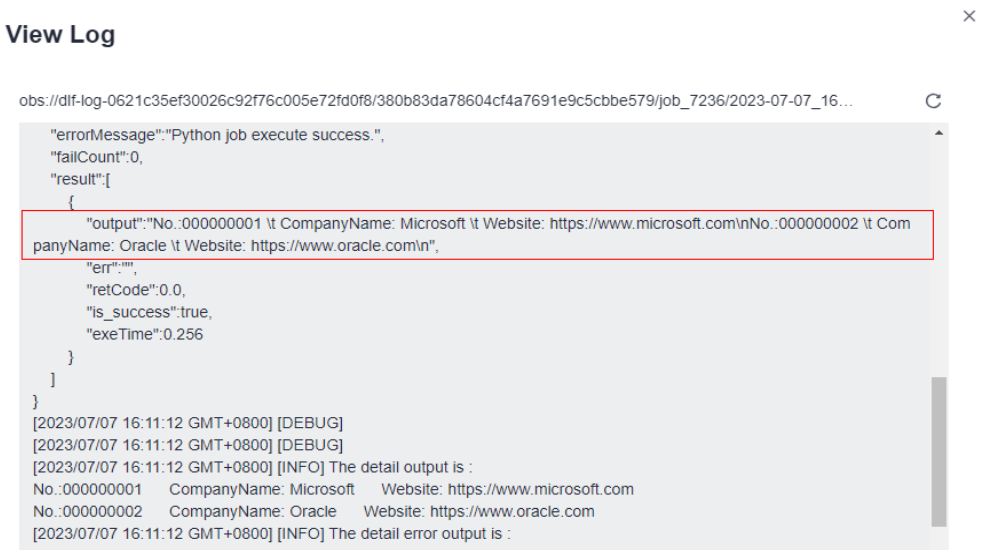
The parameters will be transferred to the Python statement when the statement is executed. The parameters are separated by spaces, for example, **Microsoft Oracle**. The parameters must be referenced by the Python statement. Otherwise, the parameters are invalid.

Figure 6-208 Configuring properties of the Python node



Step 3 Click **Test** and view the job running result.

Figure 6-209 Checking the job execution result



Step 4 Click **Save**. The job configuration is complete.

Step 5 Click **Submit**. After a version is submitted, the job can be scheduled.

----End

6.14.12 Developing a DWS SQL Job

This section describes how to use the DWS SQL node to develop a job in DataArts Factory.

Scenario

This tutorial describes how to develop a DWS job to collect the sales volume of a store on the previous day.

Preparing the Environment

- Enable DWS and create a DWS cluster for running DWS SQL jobs.
- Enable a CDM incremental package. Create a CDM cluster.

  Ensure that the VPC, subnet, and security group of the CDM cluster are the same as those of the DWS cluster so that the two clusters can communicate with each other.

Creating a DWS Data Connection

Before developing a DWS SQL job, you must create a data connection to DWS on the **Manage Data Connections** page of **Management Center**. The data connection name is **dws_link**.

The key parameters are as follows:

- **Cluster Name**: Select the DWS cluster you have created when preparing the environment.
- **Agent**: Select the CDM cluster you have created when preparing the environment.

Creating a Database

Create a **gaussdb** database by following the instructions in **Creating a Database**.

Creating Data Tables

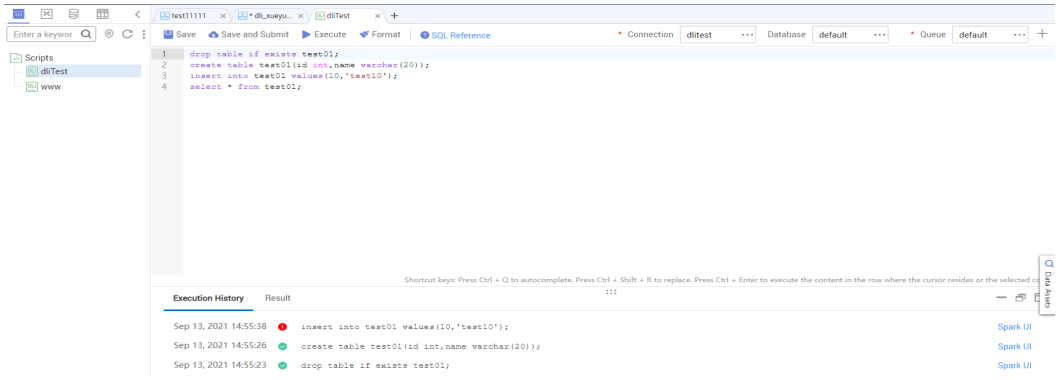
Create tables **trade_log** and **trade_report** in the **gaussdb** database. The following is an example script for creating the tables:

```
create schema store_sales;
set current_schema= store_sales;
drop table if exists trade_log;
CREATE TABLE trade_log
(
      sn          VARCHAR(16),
      trade_time   DATE,
      trade_count   INTEGER(8)
);
set current_schema= store_sales;
drop table if exists trade_report;
CREATE TABLE trade_report
(
      rq   DATE,
      trade_total   INTEGER(8)
);
```

Developing a DWS SQL Script

Choose **Development** > **Develop Script** and create a DWS SQL script named **dws_sql**. Enter an SQL statement in the editor to collect the sales amount of the previous day.

Figure 6-210 Developing a script



Key notes:

- The script development area in **Figure 6-210** is a temporary debugging area. After you close the script tab, the development area will be cleared. You can click **Submit** to save and submit a script version.
- **Connection**: Select the data connection created in **Creating a DWS Data Connection**.

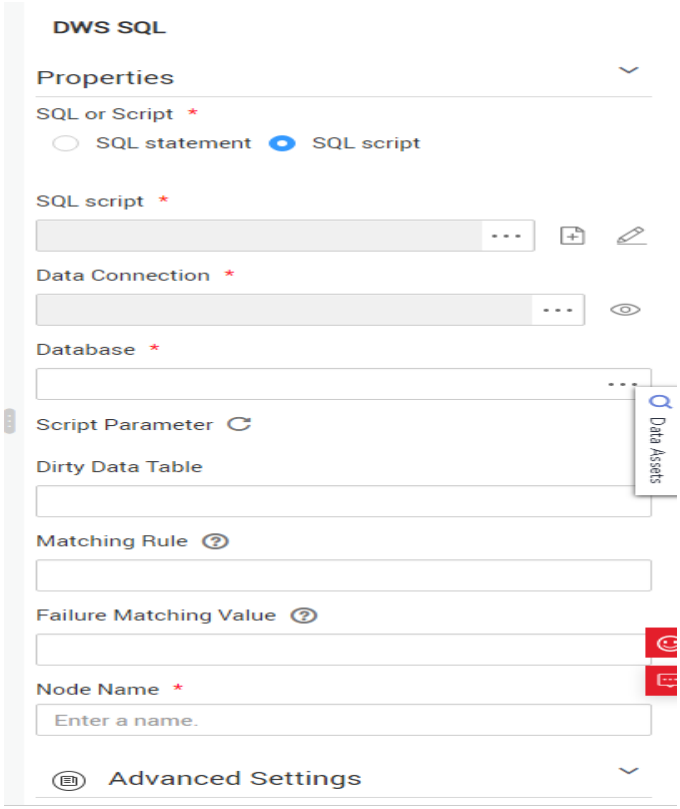
Developing a DWS SQL Job

After developing the DWS SQL script, create a job for periodically executing the DWS SQL script.

Step 1 Create a batch job named **job_dws_sql**.

Step 2 Go to the job development page, drag the DWS SQL node to the canvas, and click the node to configure its properties.

Figure 6-211 Configuring properties for the DWS SQL node



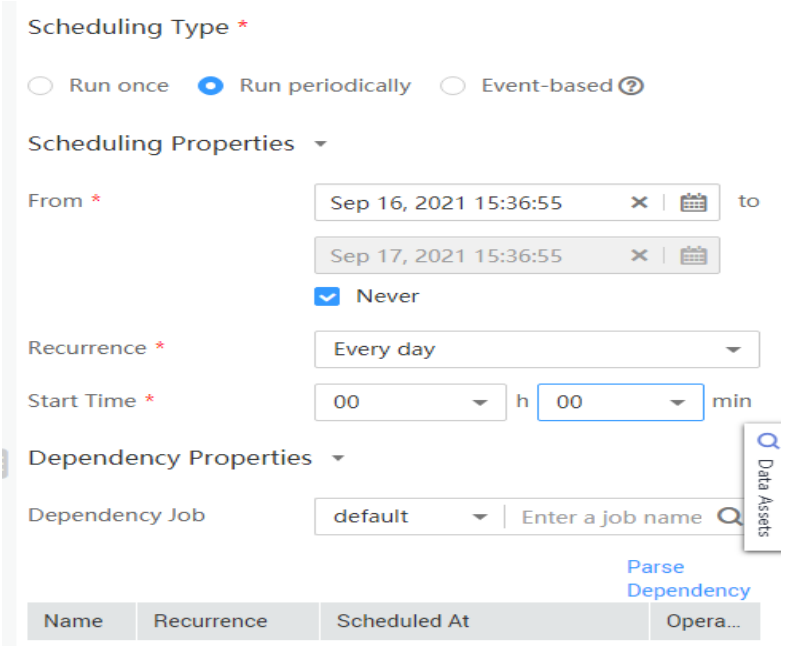
Key properties:

- **SQL script**: Associate with the **dws_sql** script developed in **Developing a DWS SQL Script**.
- **Data Connection**: Select the data connection configured in the **dws_sql** script. The data connection can be changed.
- **Database**: Select the database configured in the **dws_sql** script. The database can be changed.
- **Script Parameter**: Obtain the value of **yesterday** using the following EL expression:
  ```
  #{Job.getYesterday("yyyy-MM-dd")}
  ```
- **Node Name**: The name of the **dws_sql** script is displayed by default. The name can be changed.

Step 3 After configuring the job, click ▶ to test it.

Step 4 If the test is successful, click the blank area on the canvas and then the **Scheduling Setup** tab on the right. On the displayed page, configure the scheduling policy.

Figure 6-212 Configuring the scheduling policy



Parameter descriptions:

From Aug 6 to Aug 31 in 2021, the job was executed once at 02:00 every day.

Step 5 Click **Submit** and then **Execute**. The job will be executed automatically every day.

----End

6.14.13 Developing a Hive SQL Job

This section introduces how to develop Hive SQL scripts on DataArts Factory.

Scenario Description

As a one-stop big data development platform, DataArts Factory supports development of multiple big data tools. Hive is a data warehouse tool running on Hadoop. It can map structured data files to a database table and provides a simple SQL search function that converts SQL statements into MapReduce tasks.

Preparations

- MRS has been enabled and an MRS cluster has been created for running Hive SQL jobs.

  The MRS cluster must contain the Hive component.
- Cloud Data Migration (CDM) has been enabled. A CDM cluster has been created for providing an agent for communication between DataArts Factory and MRS.

  Ensure that the VPC, subnet, and security group of the CDM cluster are the same as those of the MRS cluster so that the two clusters can communicate with each other.

Creating a Hive Data Connection

Before developing a Hive SQL script, you must create a data connection to MRS Hive on the **Manage Data Connections** page of **Management Center**. The data connection name is **hive1009**.

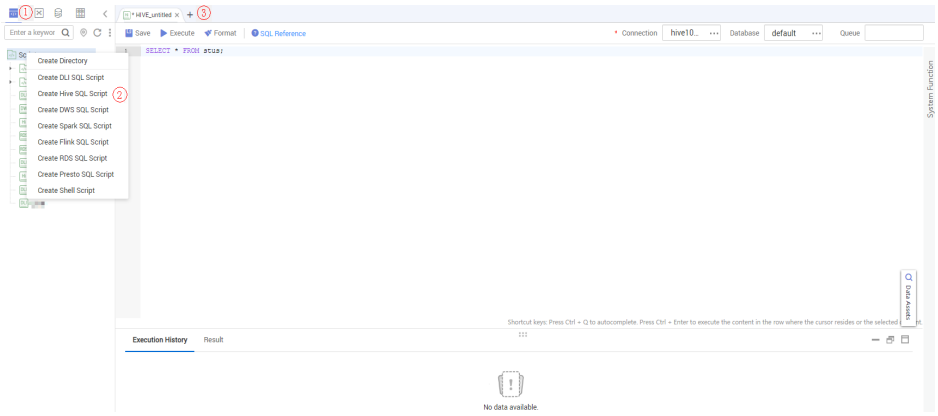
Description of key parameters:

- **Cluster Name**: Enter the name of the created MRS cluster.
- **Agent**: Select the created CDM cluster.

Developing a Hive SQL Script

Choose **Development** > **Develop Script** and create a Hive SQL script named **hive_sql**. Then enter SQL statements in the editor to fulfill business requirements.

Figure 6-213 Developing a script



Notes:

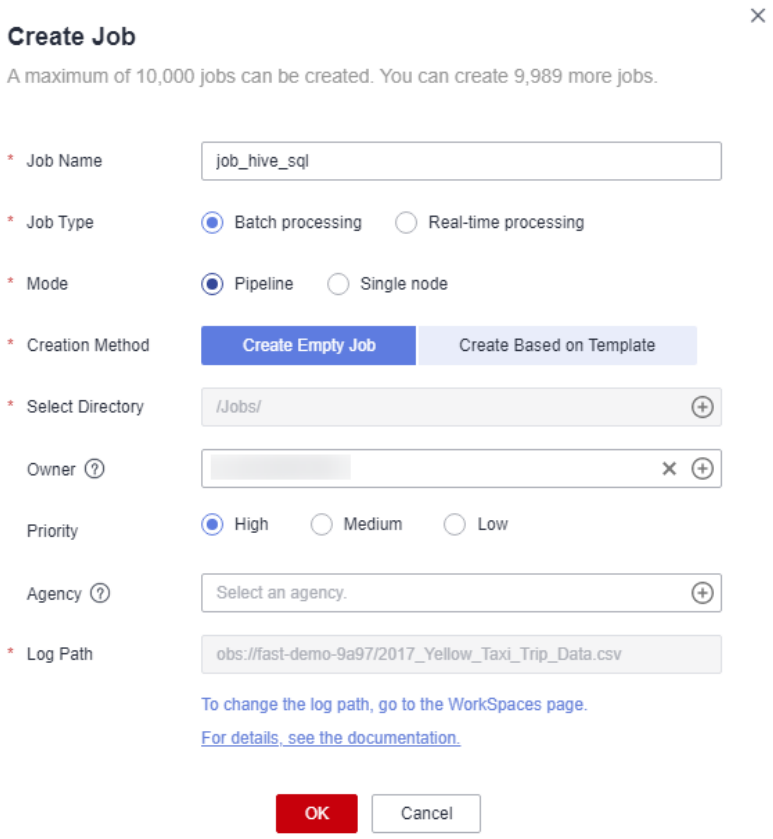
- The script development area in **Figure 6-213** is a temporary debugging area. After you close the tab page, the development area will be cleared. You can click **Submit** to save and submit a script version.
- Data Connection: Connection created in **Creating a Hive Data Connection**.

Developing a Hive SQL Job

After the Hive SQL script is developed, build a periodically deducted job for the Hive SQL script so that the script can be executed periodically.

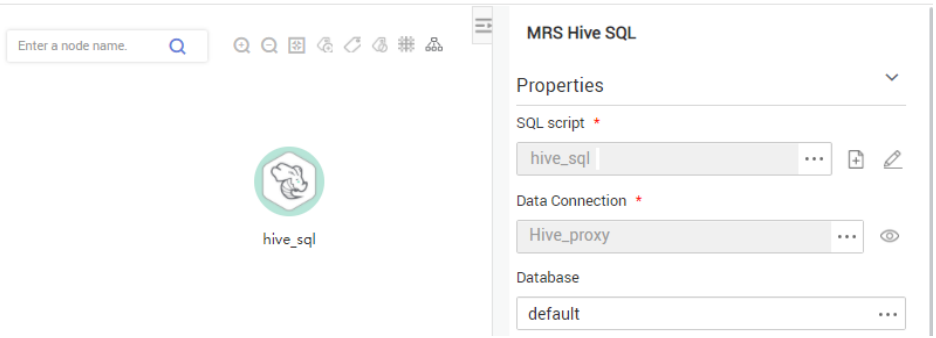
Step 1 Create an empty DataArts Factory job named **job_hive_sql**.

Figure 6-214 Creating a job named job_hive_sql



Step 2 Go to the job development page, drag the MRS Hive SQL node to the canvas, and click the node to configure node properties.

Figure 6-215 Configuring properties for an MRS Hive SQL node



Description of key properties:

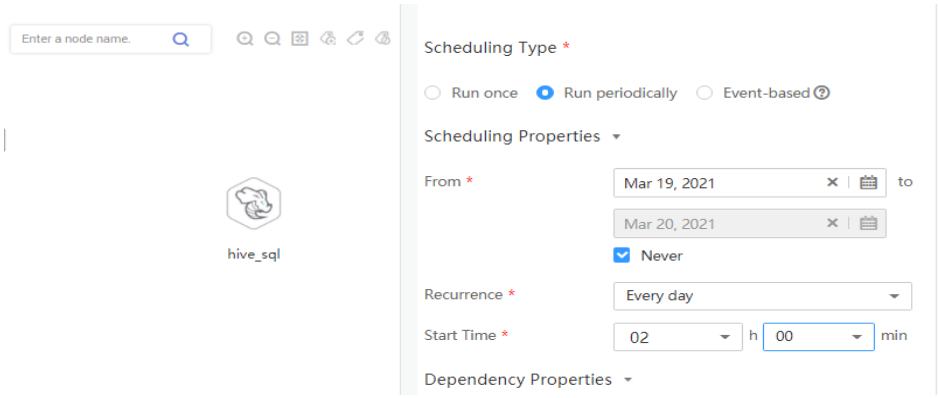
- SQL Script: Hive SQL script **hive_sql** that is developed in **Developing a Hive SQL Script**.
- Data Connection: Data connection that is configured in the SQL script **hive_sql** is selected by default. The value can be changed.
- Database: Database that is configured in the SQL script **hive_sql** and is selected by default. The value can be changed.

- Node Name: Name of the SQL script **hive_sql** by default. The value can be changed.

Step 3 After configuring the job, click ▶ to test it.

Step 4 If the job runs successfully, click the blank area on the canvas and configure the job scheduling policy on the scheduling configuration page on the right.

Figure 6-216 Configuring the scheduling mode



NOTE

The job is executed at 02:00 every day from Jan 1, 2021 to Jan 25, 2021.

Step 5 Click **Submit** and **Execute**. The job will be automatically executed every day.

----End

6.14.14 Developing a DLI Spark Job

This section introduces how to develop a DLI Spark job on DataArts Factory.

Scenario Description

In most cases, SQL is used to analyze and process data when using Data Lake Insight (DLI). However, SQL is usually unable to deal with complex processing logic. In this case, Spark jobs can help. This section uses an example to demonstrate how to submit a Spark job on DataArts Factory.

The general submission procedure is as follows:

1. Create a DLI cluster and run a Spark job using physical resources of the DLI cluster.
2. Obtain a demo JAR package of the Spark job and associate with the JAR package on DataArts Factory.
3. Create a DataArts Factory job and submit it using the DLI Spark node.

Preparations

- Object Storage Service (OBS) has been enabled and a bucket, for example, **obs://dlfexample**, has been created for storing the JAR package of the Spark job.

- DLI has been enabled, and the Spark cluster **spark_cluster** has been created for providing physical resources required for the Spark job.

Obtaining Spark Job Code

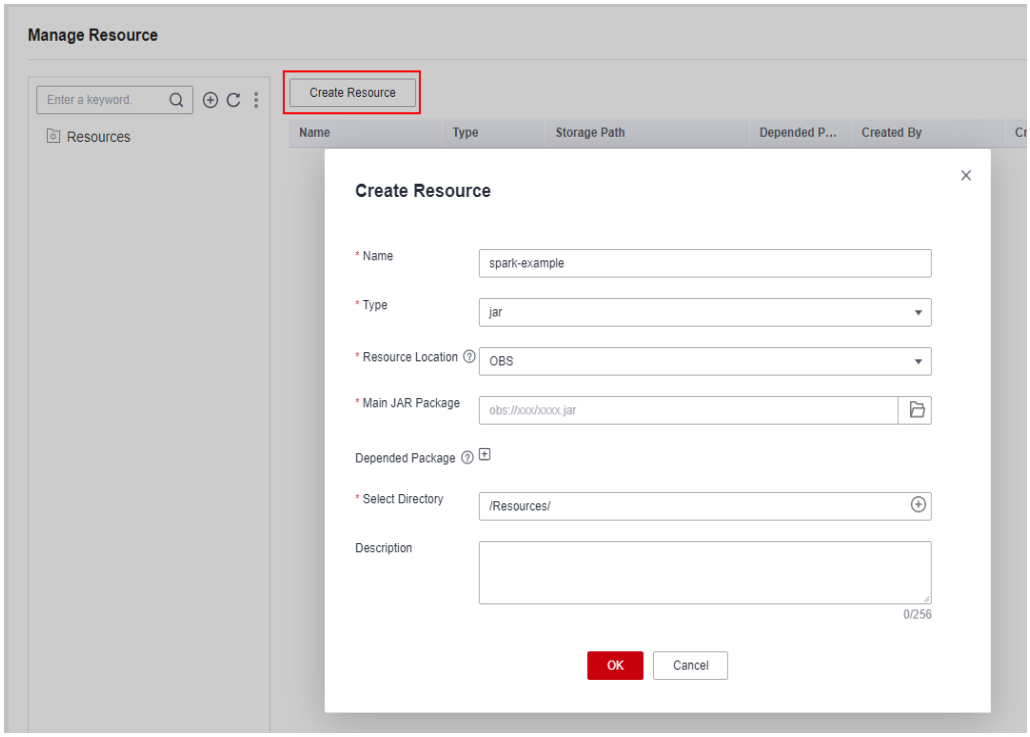
The Spark job code used in this example comes from the maven repository that can be download from **https://repo.maven.apache.org/maven2/org/apache/spark/spark-examples_2.10/1.1.1/spark-examples_2.10-1.1.1.jar**. This Spark job is to calculate the approximate value of π.

Step 1 After obtaining the JAR package of the Spark job codes, upload it to the OBS bucket. The save path is **obs://dlfexample/spark-examples_2.10-1.1.1.jar**.

Step 2 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

Step 3 In the navigation tree on the left, choose **Configuration** > **Manage Resource**. Click **Create Resource** and create resource **spark-example** on DataArts Factory and associate it with the JAR package obtained in **Step 1**.

Figure 6-217 Creating a resource



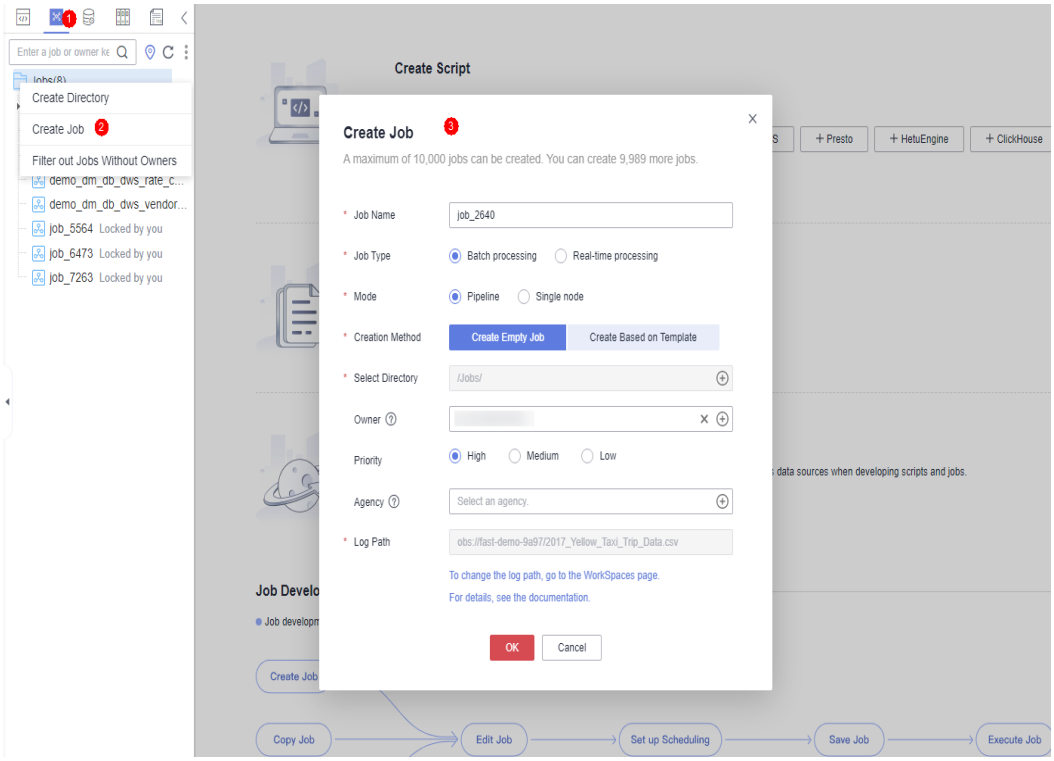
----End

Submitting a Spark Job

You need to create a job on DataArts Factory and submit the Spark job using the DLI Spark node of the job.

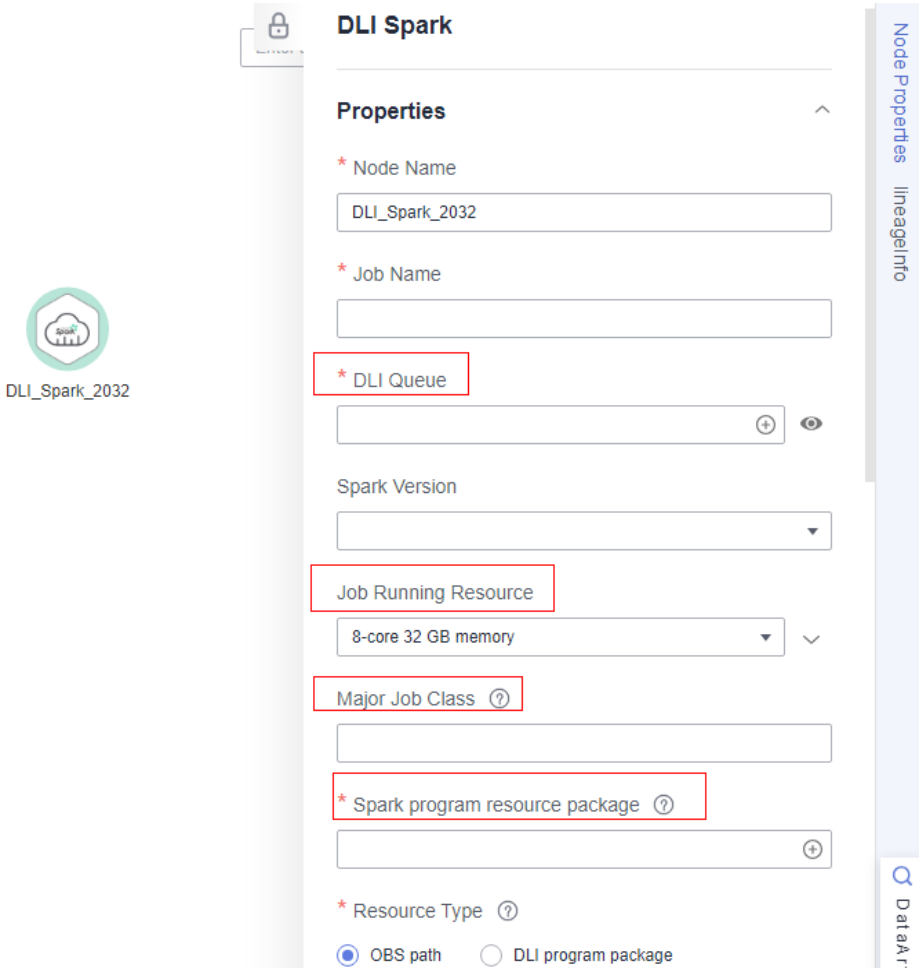
Step 1 Create a job named **job_DLI_Spark** for the DataArts Factory module.

Figure 6-218 Creating a job



Step 2 Go to the job development page, drag the DLI Spark node to the canvas, and click the node to configure node properties.

Figure 6-219 Configuring node properties

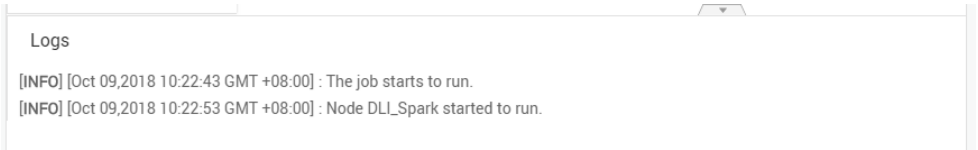


Description of key properties:

- **DLI Queue**: Select a DLI queue.
- Job Running Resource: Maximum CPU and memory resources that can be used when a DLI Spark node is running.
- Major Job Class: major class of a DLI Spark node. In this example, the major class is **org.apache.spark.examples.SparkPi**.
- **Spark program resource package**: Select the resources created in **Step 3**.

Step 3 After the job orchestration is complete, click ▷ to test the job.

Figure 6-220 Job logs (for reference only)



Step 4 If no error is recorded in logs, save and submit the job.

----End

6.14.15 Developing an MRS Flink Job

This section describes how to develop an MRS Flink job on DataArts Factory. Use an MRS Flink job to count the number of words.

Prerequisites

- You have the permission to access OBS paths.
- MRS has been enabled and an MRS cluster has been created.

Data Preparation

- Download the Flink job resource package **wordcount.jar** from **https://github.com/huaweicloudDocs/dgc/blob/master/WordCount.jar**.

  You must verify the integrity of the download Flink job resource package. In Windows, open the CLI and run the following command to generate the SHA-256 value of the downloaded JAR package. In the command, **D:\wordcount.jar** is an example local path and name of the JAR package. Replace it with the actual value.

  ```
  certutil -hashfile D:|wordcount.jar SHA256
  ```

  The following is an example command output:

  ```
  SHA-256 hash value of D:\wordcount.jar:
  0859965cb007c51f0d9ddaf7c964604eb27c39e2f1f56e082acb20c8eb05ccc4
  CertUtil: -hashfile command executed.
  ```

  Compare the SHA-256 value of the downloaded JAR package with that of the following JAR package: If they are the same, no tampering or packet loss occurred during the package download.

  SHA-256 value:
  0859965cb007c51f0d9ddaf7c964604eb27c39e2f1f56e082acb20c8eb05ccc4
- Prepare the data file **in.txt**, which contains some English words.

Procedure

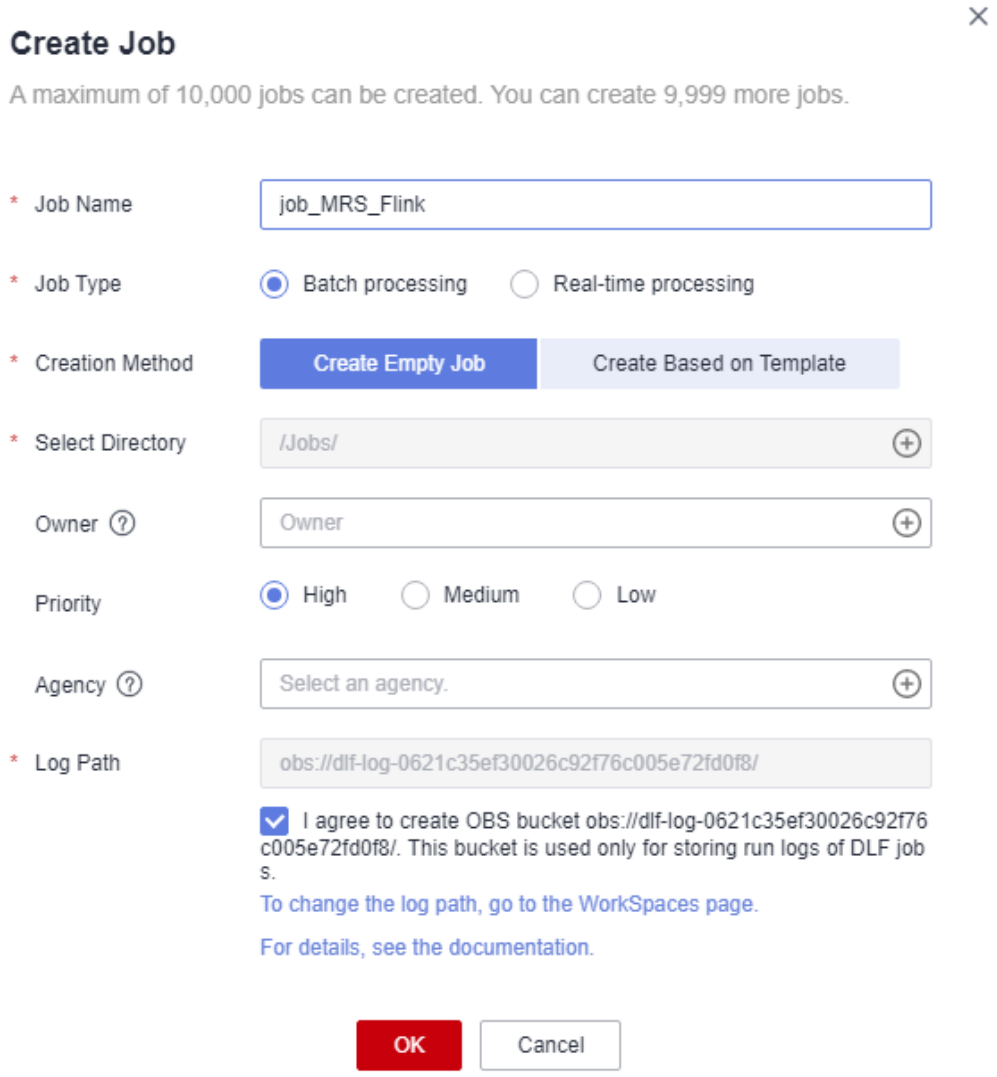
Step 1 Upload the job resource package and data file to the OBS bucket.

NOTE

In this example, upload **WordCount.jar** to **lkj_test/WordCount.jar** and **word.txt** to **lkj_test/input/word.txt**.

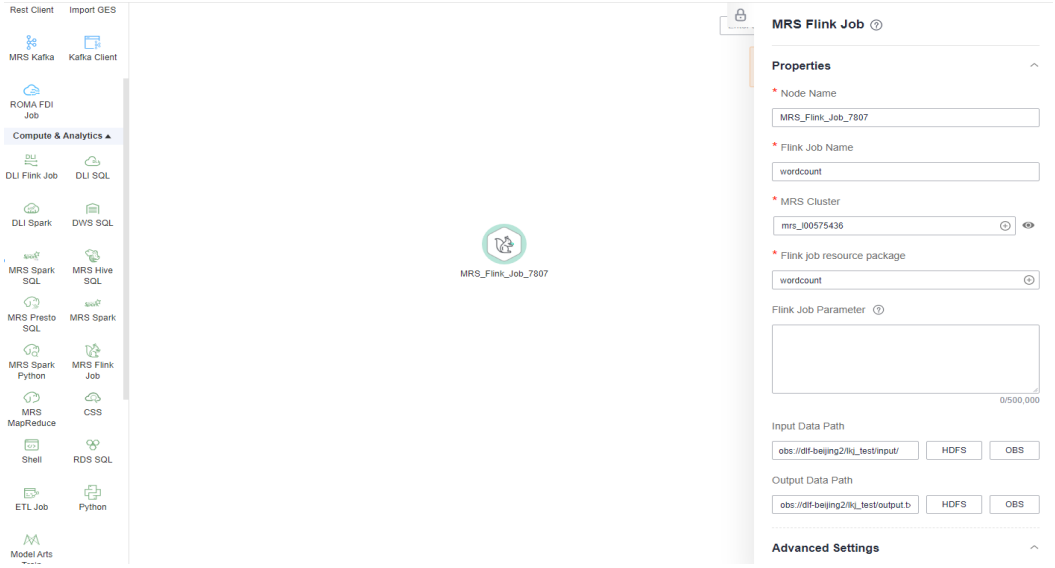
Step 2 Create an empty job named **job_MRS_Flink**.

Figure 6-221 Creating a job



Step 3 Go to the job development page, drag the **MRS Flink** node to the canvas, and click the node to configure its properties.

Figure 6-222 Configuring properties for an MRS Flink node



Parameter descriptions:

```
--Flink job name
wordcount
--MRS cluster name
Select an MRS cluster.
--Program parameter
-c    org.apache.flink.streaming.examples.wordcount.WordCount
--Flink job resource package
wordcount
--Input data path
obs://dlf-test/lkj_test/input/word.txt
--Output data path
obs://dlf-test/lkj_test/output.txt
```

Specifically:

obs://dlf-test/lkj_test/input/word.txt is the directory where the **wordcount.jar** parameters are passed. You can pass the words to count.

obs://dlf-test/lkj_test/output.txt is the directory where the output parameter file is stored. (If the **output.txt** file already exists, an error is reported.)

Step 4 Click **Test** to execute the MRS Flink job.

Step 5 After the test is complete, click **Submit**.

Step 6 Choose **Monitor Job** in the navigation pane and view the job execution result.

Step 7 View the returned records in the OBS bucket. (Skip this step if the return function is not configured.)

----End

6.14.16 Developing an MRS Spark Python Job

This section describes how to develop an MRS Spark Python on DataArts Factory.

Case 1: Using an MRS Spark Python Job to Count the Number of Words

Prerequisites

You have the permission to access OBS paths.

Data preparation

- Prepare the script file **wordcount.py** with the following content:

```
# -*- coding: utf-8 -*
import sys
from pyspark import SparkConf, SparkContext
def show(x):
    print(x)
if __name__ == "__main__":
    if len(sys.argv) < 2:
        print ("Usage: wordcount <inputPath> <outputPath>")
        exit(-1)
    # Create SparkConf.
    conf = SparkConf().setAppName("wordcount")
    # Create SparkContext. Pass the conf=conf parameter.
    sc = SparkContext(conf=conf)
    inputPath = sys.argv[1]
    outputPath = sys.argv[2]
    lines = sc.textFile(name = inputPath)
    # Split each line of data by space to obtain words.
    words = lines.flatMap(lambda line:line.split(" "),True)
    # Pair each word into a tuple count 1.
    pairWords = words.map(lambda word:(word,1),True)
    # Use three partitions (reduceByKey) for summarization.
    result = pairWords.reduceByKey(lambda v1,v2:v1+v2)
    # Print the result.
    result.foreach(lambda t :show(t))
    # Save the result to a file.
    result.saveAsTextFile(outputPath)
    # Stop SparkContext.
    sc.stop()
```

  NOTE

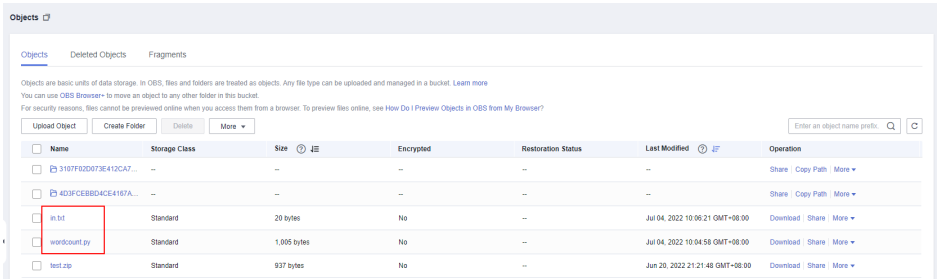
  The encoding format must be set to UTF-8. Otherwise, an error will occur during script execution.

- Prepare the data file **in.txt**, which contains some English words.

Procedure

Step 1 Upload the script and data file to the OBS bucket.

Figure 6-223 Uploading files to an OBS bucket

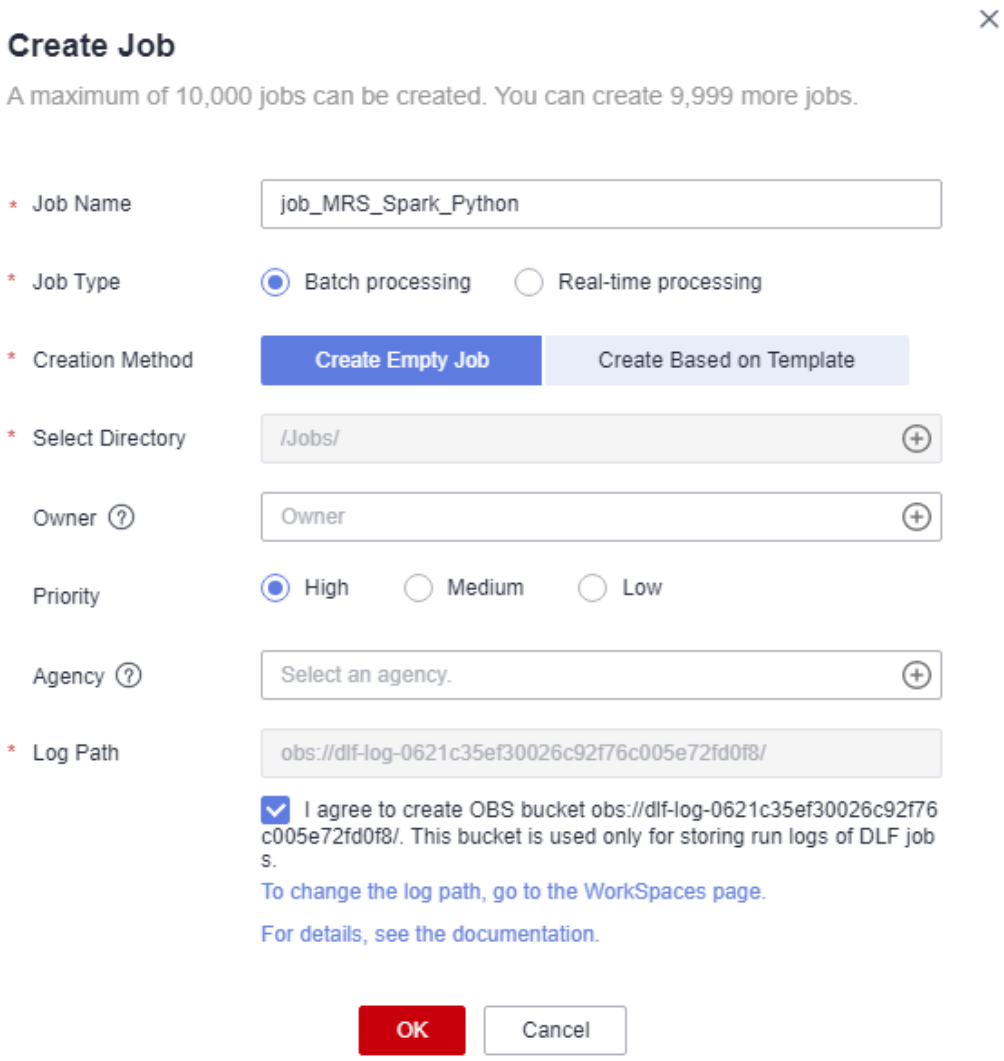


NOTE

In this example, upload **wordcount.py** and **in.txt** to **obs://obs-tongji/python/**.

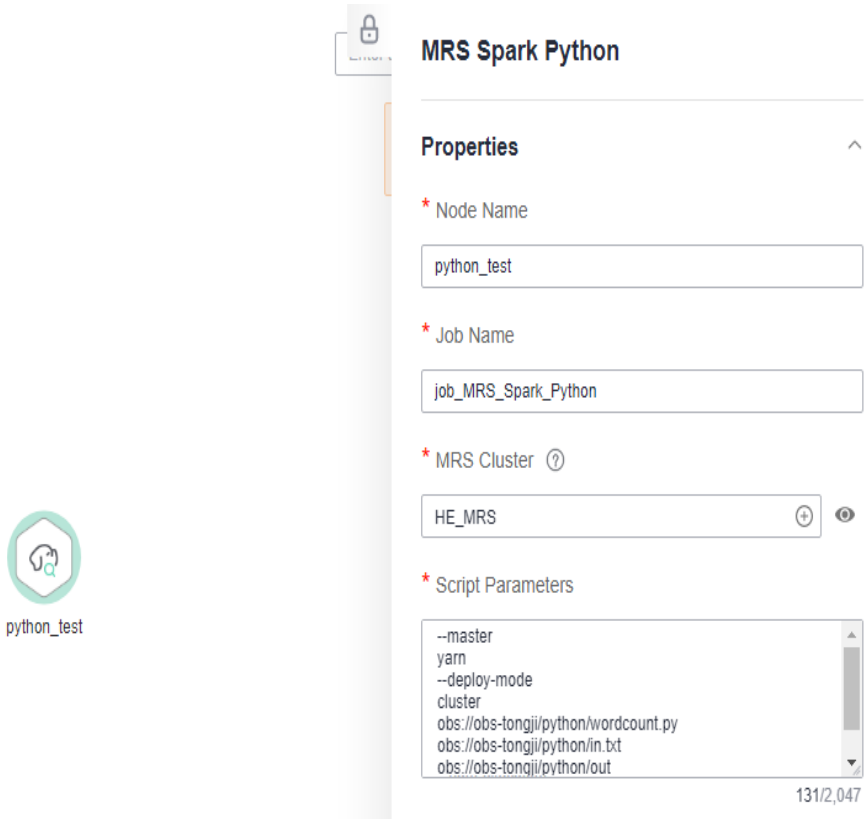
Step 2 Create an empty job named **job_MRS_Spark_Python**.

Figure 6-224 Creating a job



Step 3 Go to the job development page, drag the **MRS Spark Python** node to the canvas, and click the node to configure its properties.

Figure 6-225 Configuring properties for an MRS Spark Python node



Parameter descriptions:

```
--master
yarn
--deploy-mode
cluster
obs://obs-tongji/python/wordcount.py
obs://obs-tongji/python/in.txt
obs://obs-tongji/python/out
```

Specifically:

obs://obs-tongji/python/wordcount.py is the directory where the script is stored.

obs://obs-tongji/python/in.txt is the directory where the **wordcount.py** parameters are passed. You can pass the words to count.

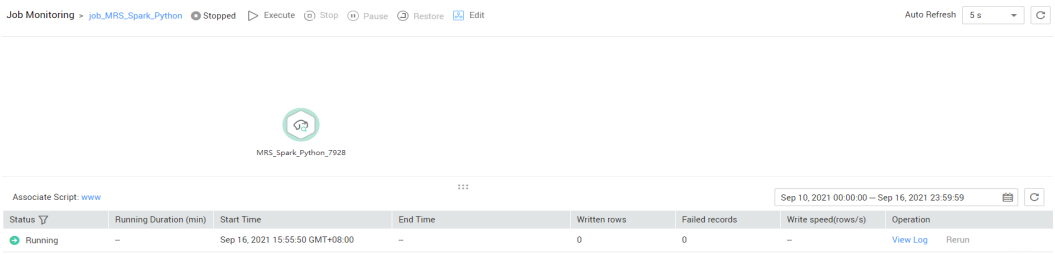
obs://obs-tongji/python/out is the directory where output parameters are stored. This directory will also be created in the OBS bucket automatically. If the **out** directory already exists in the OBS bucket, an error will occur.

Step 4 Click **Test** to execute the script job.

Step 5 After the test is complete, click **Submit**.

Step 6 Choose **Monitor Job** in the navigation pane and view the job execution result.

Figure 6-226 Viewing the job execution result



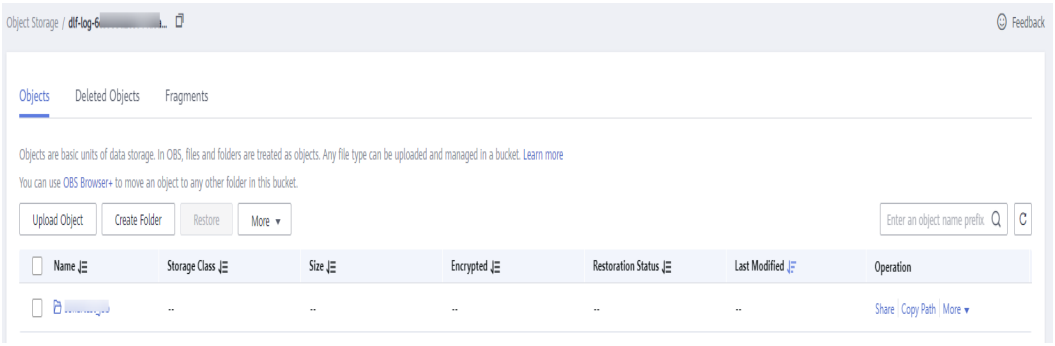
The job log shows that the job was successfully executed.

Figure 6-227 Job run logs

Figure 6-228 Job execution status

Step 7 View the returned records in the OBS bucket. (Skip this step if the return function is not configured.)

Figure 6-229 Viewing the returned records in the OBS bucket



----End

Case 2: Using an MRS Spark Python Job to Print hello python

Prerequisites

You have the permission to access OBS paths.

Data preparation

Prepare the script file **zt_test_sparkPython1.py** with the following content:

```
from pyspark import SparkContext, SparkConf
conf = SparkConf().setAppName("master"). setMaster("yarn")
sc = SparkContext(conf=conf)
print("hello python")
sc.stop()
```

Procedure

Step 1 Upload the script file to an OBS bucket.

Step 2 Create an empty job.

Step 3 Go to the job development page, drag the **MRS Spark Python** node to the canvas, and click the node to configure its properties.

Parameter descriptions:

```
--master
yarn
--deploy-mode
cluster
obs://obs-tongji/python/zt_test_sparkPython1.py
```

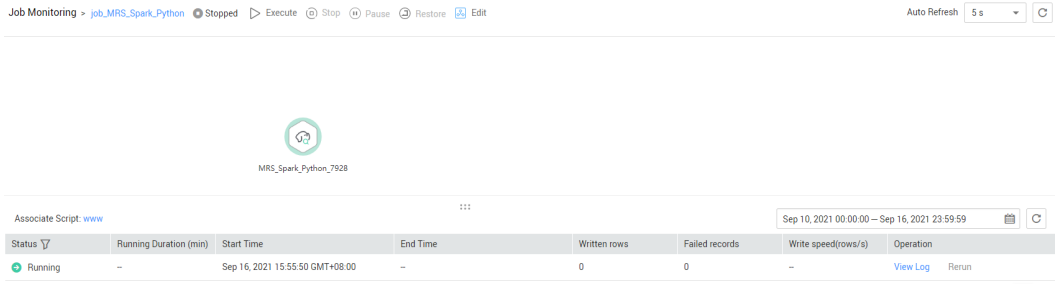
zt_test_sparkPython1.py indicates the directory where the script is stored.

Step 4 Click **Test** to execute the script job.

Step 5 After the test is complete, click **Submit**.

Step 6 Choose **Monitor Job** in the navigation pane and view the job execution result.

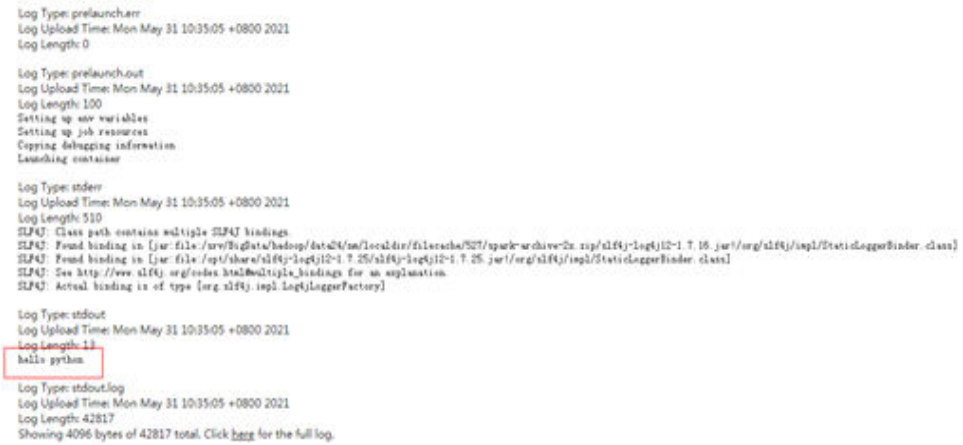
Figure 6-230 Viewing the job execution result



Step 7 Verify the log.

Log in to MRS Manager and check that the log on YARN contains **hello python**.

Figure 6-231 Viewing logs on YARN



----End

6.14.17 More Cases for Reference

For more advanced guidance and cases of DataArts Factory, see **Best Practices**.

7 DataArts Quality

7.1 Metric Monitoring (Unavailable Soon)

7.1.1 Overview

NOTICE

The metric monitoring function of DataArts Quality will be unavailable soon. You are advised to use DataArts Architecture in the future, which provides comprehensive metric design and management capabilities.

The Metric Monitoring module manages business metrics.

To monitor a business metric, customize a SQL metric, define a rule based on the logical expression of the metric, and create and run a business scenario. Based on the running result of the business scenario, you can determine whether the business metric meets the quality rule. The running result of the business scenario may be any of the following:

- **Normal**: The instance stops normally and the running result meets the expectation.
- **Alarming**: The instance stops normally, but the running result does not meet the expectation.
- **Abnormal**: The instance stops unexpectedly.
- **--**: The instance is running, but no running result is displayed.

The following table describes modules under **Quality Monitoring**.

Function	Description
Dashboard	Default homepage. This page contains the following parts: • **Quick Start** that demonstrates how you can use metric monitoring • Running and alarm statuses for the business scenario instance over the last seven days • Alarms, scenarios, and metrics in different time periods
Metrics	You can create metrics on this page.
Rules	You can create rules based on the logical expressions of metrics on this page.
Scenarios	A business scenario can be considered as a business metric quality job. On this page, you can schedule and run a created rule group.
O&M	You can view the running statuses of business scenario instances and handle O&M issues. The **Subscriptions** page displays the running statuses of all the tasks you have subscribed to.

7.1.2 Creating a Metric

You can manage all business metrics, including the metric sources and definitions. Business metrics are stored in directories.

Metrics in DataArts Quality are independent of business metrics and technical metrics in DataArts Architecture.

Prerequisites

You have created a home directory. To create a home directory, log in to the DataArts Studio console, click **Access** under a specific instance, choose **More** > **DataArts Quality** on the **Workspaces** tab page, choose **Metric Monitoring** > **Metric Management** from the left navigation bar on the page displayed, and create a directory. Before creating a metric for a data connection, select a directory to store the metric. For details, see **Figure 7-1**.

Figure 7-1 Directory that stores the metric to create

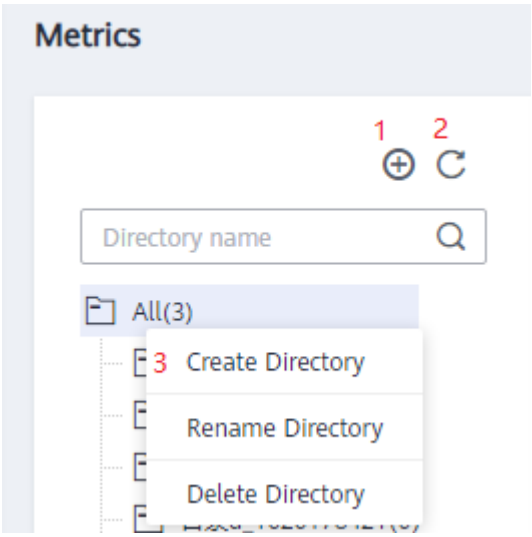


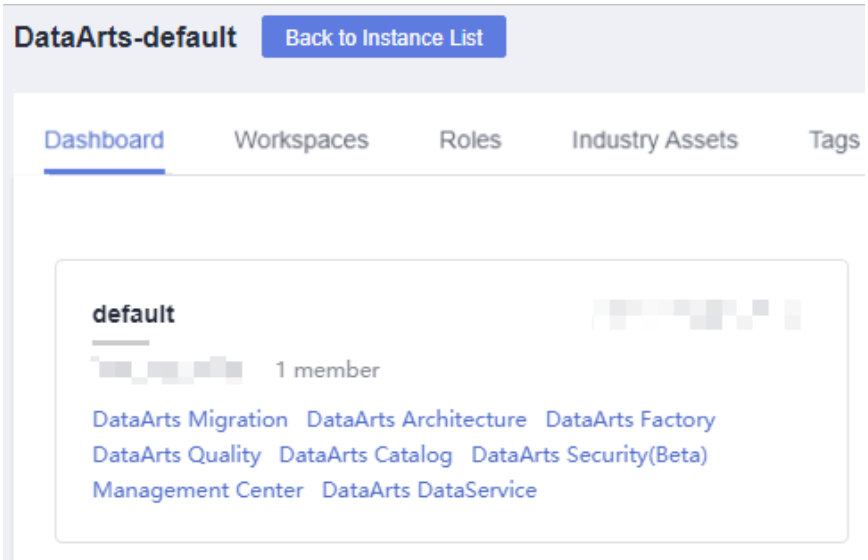
Table 7-1 Buttons in the navigation bar

No.	Description
1	Create Directory
2	Refresh Directory
3	Right-click **All** to create, rename, or delete a directory.

Creating a Metric

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Quality**.

Figure 7-2 DataArts Quality



2. Choose **Metric Monitoring** > **Metrics** from the left navigation bar.
3. Click **Create**. In the dialog box displayed, set the parameters based on **Table 7-2**.

Table 7-2 Metric parameters

Parameter	Description
Metric Name	The name of a metric, which contains 1 to 64 characters and consists of only letters, numbers, and underscores (_).
Data Connection	Select a created data connection from the drop-down list box. **NOTE** • Currently, only DWS, PostgreSQL, MRS Hive, DLI, MRS ClickHouse, and MySQL are supported. • Metrics are closely connected based on data connections. Therefore, you must establish data connections in the metadata management module before creating metrics.
Database/ Queue	Select the database where the metric runs. **NOTE** If DLI is selected as the data connection, a running queue is required.
Description	Information to better identify a metric. It cannot exceed 4096 characters.
Directory	Directory for storing metrics. You can select a created directory. **Figure 7-1** shows the directory.
Metric Type	**Custom** is supported. You can customize an SQL statement to define the metric source.

7.1.3 Creating a Rule

You can manage all rules that define relationships between metrics or between metrics and values. Rules are stored in directories.

Prerequisites

You have created a home directory. To create a home directory, log in to the DataArts Studio console, click **Access** under a specific instance, choose **More** > **DataArts Quality** on the **Workspaces** tab page, choose **Metric Monitoring** > **Rule Management** from the left navigation bar on the page displayed, and create a directory. Before creating a rule for a metric, select a directory to store the rule. For details, see **Figure 7-3**.

Figure 7-3 Directory that stores the rule to create

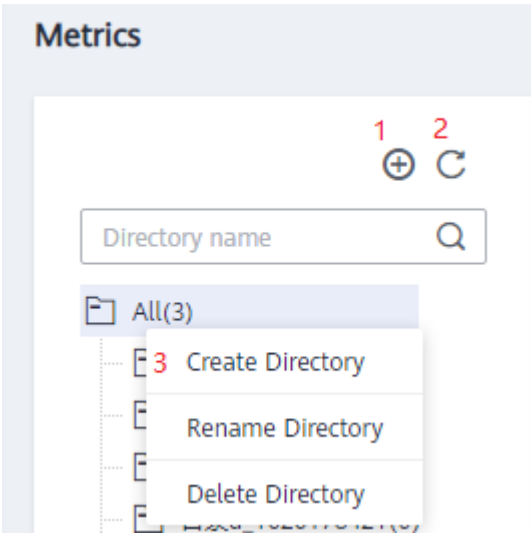


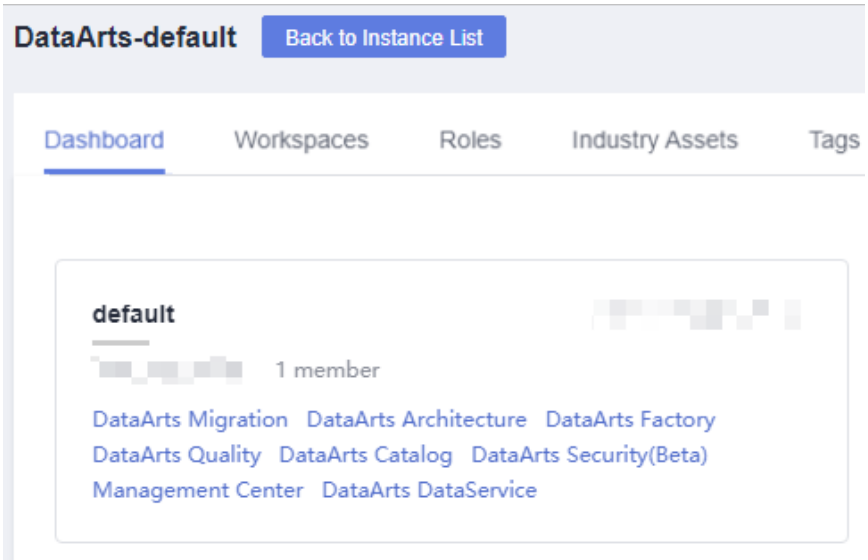
Table 7-3 Buttons in the navigation bar

No.	Description
1	Create Directory
2	Refresh Directory
3	Right-click **All** to create, rename, or delete a directory.

Creating a Rule

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Quality**.

Figure 7-4 DataArts Quality



2. Choose **Metric Monitoring** > **Rule Management** from the left navigation bar.
3. Click **Create**. In the dialog box displayed, set the parameters based on **Table 7-4**.

Table 7-4 Rule parameters

Parameter	Description
Rule Name	The name of a rule, which contains 1 to 64 characters and consists of only letters, numbers, and underscores (_).
Description	Information to better identify a rule. It cannot exceed 4096 characters.
Directory	The directory that stores the rule. You can select a created directory. **Figure 7-3** shows the directory.
Define Relationship	A relationship is a logical expression between a metric and a value or between metrics. The relationship can contain arithmetic operations. Metrics are abbreviated to lowercase letters a to z and are added in the alphabetic order of metric abbreviations. **NOTE** Only one valid logical expression and the simple four arithmetic operations are supported.

7.1.4 Creating a Scenario

You can manage all scenarios that define the logical relationships between rules. Scenarios are stored in directories.

Prerequisites

You have created a home directory. To create a home directory, log in to the DataArts Studio console, click **Access** under a specific instance, choose **More** > **DataArts Quality** on the **Workspaces** tab page, choose **Metric Monitoring** > **Business Scenario Management** from the left navigation bar on the page displayed, and create a directory. Before creating a scenario for rules, select a directory to store the scenario. For details, see **Figure 7-5**.

Figure 7-5 Directory that stores a scenario

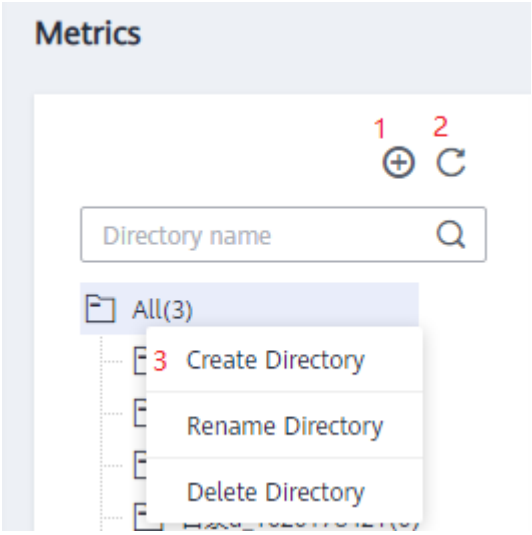


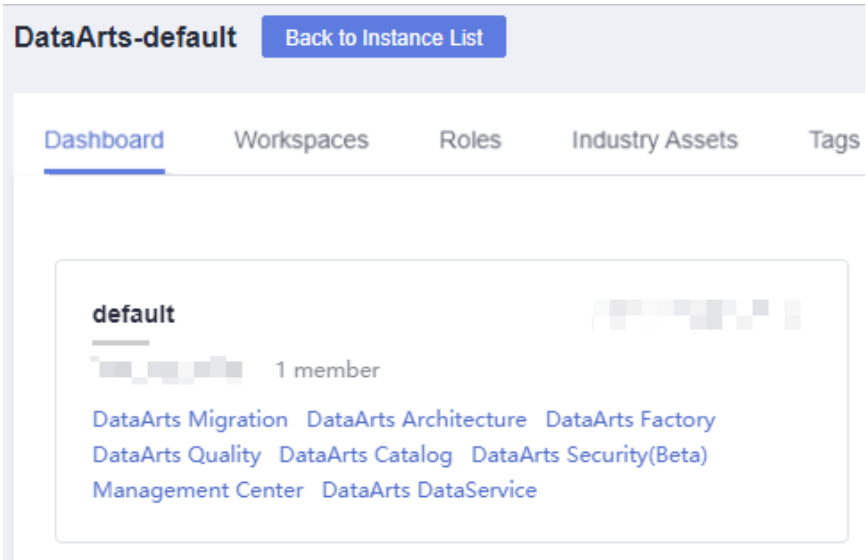
Table 7-5 Buttons in the navigation bar

No.	Description
1	Create Directory
2	Refresh Directory
3	Right-click **All** to create, rename, or delete a directory.

Creating a Scenario

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Quality**.

Figure 7-6 DataArts Quality



2. Choose **Metric Monitoring** > **Business Scenario Management** from the left navigation bar.
3. Click **Create**. In the dialog box displayed, set the parameters based on **Table 7-6**.

Table 7-6 Scenario parameters

Parameter	Description
Basic Configuration	
Scenario Name	The name of a scenario, which contains 1 to 64 characters and only consists of letters, numbers, and underscores (_).
Description	Information to better identify a scenario. It cannot exceed 256 characters.
Directory	The directory that stores the scenario. You can select a created directory. **Figure 7-5** shows the directory.
Business Level	The options are **Warning**, **Minor**, **Major**, and **Critical**. The business level determines the template for sending notification messages.
Rule Group Configuration	
Define Rule Group	Group of rules. Logical expressions are used between rules.
Rule A	You can select a rule from the drop-down list. You can also click ⊕ to add multiple rules.
Subscription Configuration	
Notification	Set this to  or  to enable or disable the notification function.
Notification Type	The options are as follows: • Trigger alarms • Run successfully
Topic	Select a message notification topic. **NOTE** Currently, only SMS and email are available for subscribing to topics.

4. Click **Next** to go to the page where you can select a scheduling mode. Currently, **Schedule once** and **Schedule periodically** are supported. Set parameters for scheduling periodically by referring to **Table 7-7**.

Table 7-7 Scheduling parameters

Parameter	Description
Effective	The period during which a scheduling task takes effect.
Scheduling Cycle	The frequency at which a scheduling task is executed. Related parameters are: • Minute • Hour • Day • Week
Time Interval	Interval for two consecutive scheduling tasks.
Start from	Start time and end time of the scheduling task

7.1.5 Viewing a Scenario Instance

You can manage all scenarios, view metric running statuses, query run logs, and handle issues on the **O&M Management** page.

GUI Description

The following figure shows the areas and buttons on the **O&M** page.

Figure 7-7 O&M Management page

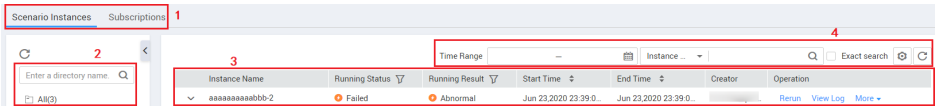


Table 7-8 Entry

No.	Area	Description
1	Menu bar	The menu bar on the **O&M Management** page includes **Scenario Instances** and **Subscriptions**. • The **Scenario Instances** tab page lists all scenario instances that you have created. • The **Subscriptions** tab page lists all scenarios that you have subscribed. **Notification Status** is available only on the **Subscriptions** tab page. **Notification Status** indicates whether the running result of a scenario instance is subscribed to, for example, sending an alarm email.

No.	Area	Description
2	Navigation bar	Contains the directories that store scenario instances. You can store scenarios in different directories. The number next to each directory indicates the number of scenarios stored in that directory.
3	List of scenario instances	Displays the instance name, running status, and running result.
4	Search area	• Displays scenario instances selectively. For example, you can display scenario instances for a specified time range. • Displays a list of instances according to the handler, creator, or instance name. Fuzzy search is supported.

Table 7-9 Scenario instance parameters

Parameter	Description
Running Status	Displays the running status of a scenario instance. • **Successful**: The instance is successfully executed. • **Failed**: The instance fails to run. • **Running**: The instance is running.
Running Result	Displays whether the scenario instance is running properly. • **Normal**: The instance stops normally and the running result meets the expectation. • **Alarming**: The instance stops normally, but the running result does not meet the expectation. • **Abnormal**: The instance stops unexpectedly. • **--**: The instance is running, but no running result is displayed.
Rerun	Allows you to run the scenario instance again.
View Log	Allows you to view the running details of the scenario instance.
More > Resolve Issue	Allows you to perform further processing on the scenario instance. You can **Provide handling suggestions**, **Close the issue**, or **Transfer to others**. The above operations can be performed only when you are the handler of the instance.
More > View Processing Log	Allows you to view historical processing records.

7.2 Monitoring Data Quality

7.2.1 Overview

DataArts Quality is a type of quality management tool used to manage the quality of data in databases. You can filter out unqualified data in a single column or across columns, rows, and tables from the following perspectives: integrity, validity, timeliness, consistency, accuracy, and uniqueness. DataArts Quality can monitor offline data. When offline data changes, DataArts Quality verifies the data and blocks the production link to avoid the spread of the problem data. DataArts Quality also manages historical verification results so that you can analyze and grade data quality.

It can also automatically generate standardized quality rules based on the data standards in DataArts Architecture, and periodically monitor data.

The following table describes modules under **Quality Monitoring**.

Module	Description
Dashboard	The dashboard is the homepage that displays alarming and blocking information of tables. The following information is included: ● Number of jobs, instances, and anomaly tables; distributions and changes of instance running statuses in a selected period. ● Statistics about alarm classifications and table alarms of the current day, as well as the alarm trend and rule quantity of the latest seven days.
Rule Template	Rule template is a major function of DataArts Quality. You can configure rules on the **Rule Template** page. It mainly manages functions related to rule configuration and provides built-in and custom templates.
Quality Job	Quality jobs can apply rule templates or custom rules to tables for data monitoring.
Comparison Job	You can create comparison jobs to apply the created rules to two existing tables to monitor their data and output the comparison results.
O&M Management	You can view the running status of rules and handle O&M problems.
Quality Report	The system automatically generates quality reports based on the job execution result.

7.2.2 Creating Rule Templates

DataArts Quality can monitor offline data, in which quality rules play a vital role. There are 34 built-in rule templates, such as database-level, table-level, field-level, cross-source, and cross-field rule templates.

Table 7-10 System built-in rule templates

Rule Type	Dimension	Template	Description
Database-level	Integrity	Database null value scan	Calculates the number of rows with empty field values in each table in the database. The result is displayed by field.
Table-level	Accuracy	Table rows	Calculates the number of rows in a data table.
	Integrity	Data table null value scan	Calculates the number of rows with empty field values in each table. The result is displayed by field.
	Validity	Fluctuation rate in the last day	Calculates the size, field groups, and related fluctuation rate of a data table in the last day.
		Fluctuation rate in the last seven days	Calculates the size, field groups, and related fluctuation rate of a data table in the last seven days.
		Fluctuation rate in the last 30 days	Calculates the size, field groups, and related fluctuation rate of a data table in the last 30 days.
Field-level	Uniqueness	Field with a unique value	Calculates the number of rows in a data table in which a specified field has a unique value.
		Field with duplicate values	Calculates the number of rows in a data table in which a specified field has duplicate values. If the field has multiple different duplicate values, the total number of duplicate values is the calculation result.
		Unique combination of multiple fields	Checks whether the combination of multiple fields in a DWS, DLI, Hive, and SparkSQL table is unique. A maximum of 10 fields can be combined.
		Multi column uniqueness verification ignore null	Checks whether the combination of multiple fields in a DWS, DLI, Hive, and SparkSQL table is unique. A maximum of 10 fields can be combined. Null values are counted in valid rows.

Rule Type	Dimension	Template	Description
	Integrity	Field with a null value	Calculates the number of rows in a data table in which a specified field has a null value.
	Accuracy	Average field value	Calculates the average value of a specified field in a data table.
		Total field values	Calculates the total values of a specified field in a data table.
		Maximum field value	Calculates the maximum value of a specified field in a data table.
		Minimum field value	Calculates the minimum value of a specified field in a data table.
		Field length verification	Checks whether the length of a field in a DWS and a DLI table is within the allowed range.
		Field value range verification	Checks whether the value of a field in a DWS and a DLI table is within the allowed range.
		Field time verification	Checks whether the time of a field in a DWS and a DLI table is within the allowed range. Currently, only fields of the date and timestamp types are supported. Fields of the time type is not supported.
	Effectiveness	ID card verification	Checks validity of a specified field in a data table based on built-in regular expression rules. If the field is empty, it is invalid.
		Mailbox verification	Checks validity of a specified field in a data table based on built-in regular expression rules.
		Regular expression verification	Checks validity of a specified field in a data table based on a custom regular expression.
		IP address verification	Checks validity of a specified field in a data table based on built-in regular expression rules.
		Phone number format verification	Checks validity of a specified field in a data table based on built-in regular expression rules.

Rule Type	Dimension	Template	Description
		Postal code format verification	Checks validity of a specified field in a data table based on built-in regular expression rules.
		Date format verification	Checks validity of a specified field in a data table based on built-in regular expression rules.
		Validity verification	Checks validity of a specified field in a data table based on a custom regular expression.
		Enumerated value verification	Checks validity of a specified field in a data table based on a custom enumerated value.
		Ignoring of null values in enumerated value verification	Checks validity of a specified field in a DWS/DLI data table based on a custom enumerated value. Null values are counted in valid rows.
		Ignoring of null values in regular expression verification	Checks validity of a specified field in a DWS/DLI data table based on a custom regular expression. Null values are counted in valid rows.
		Ignoring of case in enumerated value verification	Checks validity of a specified field in a DWS/DLI data table based on a custom enumerated value. Case-sensitive values are counted in valid rows.
		Ignoring of null values and case in enumerated value verification	Checks validity of a specified field in a DWS/DLI data table based on a custom enumerated value. Null and case-sensitive values are counted in valid rows.
Cross-field level	Consistency	Field consistency verification	Checks whether the value of a specified field in a data table is the same as that of the reference field from the same source.
	Accuracy	Cross-field time verification	Checks whether the time relationship between a specified field in a DWS and a DLI table and the reference field meets the expectation. Currently, only fields of the date and timestamp types are supported. Fields of the time type is not supported.

Rule Type	Dimension	Template	Description
Cross-source level	Consistency	Cross-source field consistency verification	Checks consistency between different fields from different data sources based on a Hetu connection. (This system template depends on the Hetu connection and is unavailable currently.)

You cannot edit built-in rule templates or view their release history.

If the built-in rule templates do not meet your requirements, you can create rules in either of the following ways:

NOTE

Developers cannot randomly modify custom rule templates because they may be used by many users. To modify custom rule templates, contact the administrator.

- Custom template: Choose **Quality Monitoring** > **Rule Templates** and click **Create**. The created rule template is automatically allocated the corresponding rule type (table level, field level, cross-field level, or multi-table and multi-field). The template type is custom. When creating a quality/comparison job, you can select **Table rule**, **Field rule**, **Cross-field rule**, or **Multi-table and multi-field rule** for **Rule Type**, and then you can select a custom template which supports export of abnormal data but does not support quality scoring.
- Custom rule: When creating a quality job, set **Rule Type** to **Custom rule** and enter an SQL statement to define how to monitor the quality of data objects.

  NOTE

  An SQL statement can contain multiple tables in the same database, but not tables in different databases.

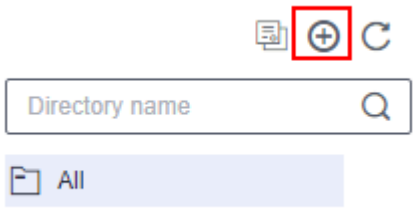
This section describes how to create a rule using a custom template. For details about how to create a custom rule, see **Creating Quality Jobs**.

Step 1 (Optional) In the left navigation pane, choose **Quality Monitoring** > **Rule Templates** and create a directory. If a directory exists, you do not need to create one. Note that rule templates, quality jobs, and comparison job are in the same directory.

Currently, you can create a directory using either of the following methods:

Click ⊕ and enter a directory name in the displayed dialog box. In this way, you can create a maximum of seven layers of directories.

Figure 7-8 Creating a directory



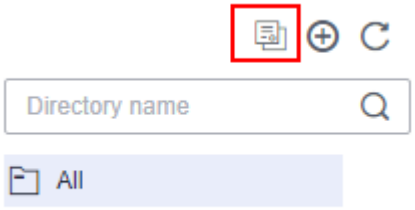
You can also click to synchronize the **subjects in DataArts Architecture** as directories. (Only published subjects can be synchronized.) The synchronized directories are consistent with the published subjects in DataArts Architecture and are displayed by layer, such as L1 and L2.

NOTE

1. The directories you create are not affected by the synchronization. If the name of a created directory conflicts with that of a subject:
   - If they conflict during the first synchronization, a subject layer (such as L1 and L2) is added to the name of the directory.
   - If they conflict after the subject is modified, the synchronization fails.
2. Changes to subjects or subject layers in DataArts Architecture cannot be automatically synchronized. You must click again to synchronize them.

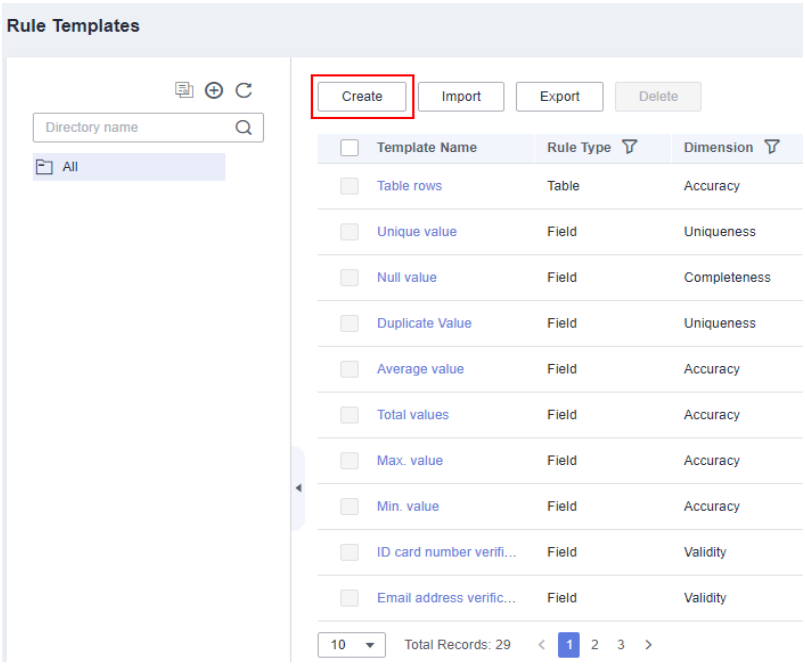
   If a subject or subject layer in DataArts Architecture is deleted and synchronized to DataArts Quality, the corresponding directory will not be deleted. Instead, the subject attributes will be deleted from the directory.
3. After the synchronization is complete, the system automatically displays the synchronization result details. You can view the names of the subjects that fail to be synchronized.

Figure 7-9 Synchronizing subjects from DataArts Architecture



Step 2 On the **Rule Templates** page, click **Create**.

Figure 7-10 Rule Template page



Step 3 In the dialog box displayed, enter the rule template name, select the rule matching dimension, define the SQL template, and describe the output result.

- **Dimension**: You can complete single-column, cross-column, cross-row, and cross-table analysis from six dimensions: completeness, validity, timeliness, consistency, accuracy, and uniqueness. When customizing a quality rule, select a dimension for rule matching.
- **Directory**: Select the directory where the rule template is located.
- **Description**: Enter the description of the custom template.
- **Relationship**: Enter an SQL statement to search for data. **${Schema_Table1}** is the table selected for the quality/comparison job. **${Column1}** is the field selected in **${Schema_Table1}**. **${Schema_Table2}** exists only when a cross-field rule is defined and indicates the reference table selected for the quality job. **${Column2}** is the field selected in **${Schema_Table2}**. The system can verify the semantics of the relationship.

  NOTE

  If you enter non-digit characters for the relationship, only the execution result is generated. Four arithmetic operations and logical operations cannot be performed, and absolute values cannot be calculated.

  A custom SQL expression must meet the following requirements:

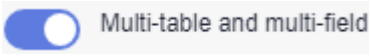
  a. A relational expression supports a maximum of five columns.

  b. The input parameters of a maximum of two tables and two fields are supported. Note: **${Column1}** is the input parameter of **${Schema_Table1}**, and **${Column2}** is the input parameter of **${Schema_Table2}**. They are specified by built-in logic.

  c. Table aliases are not supported.

  d. If multiple rows are found, only the data in the first row is used.

For example, to count the number of rows in a table, enter **select count(${Column1}) from ${Schema_Table1}**. The value of **${Column1}** is generated by clicking **Add Field Parameter**, and the value of **${Schema_Table1}** is generated by clicking **Add Database/Table Parameter**.

Click 

and enable **Add Input Parameter** to flexibly configure input parameters in SQL statements.

For example, if a field matches the number of rows in the configuration table, enter **select count(1) from ${Schema_Table1} where ${Column1} regexp ${Input_String1}**. Click **Add Field Parameter** to generate **${Column1}** and click **Add Database/Table Parameter** to generate **${Schema_Table1}**.

NOTE

When creating a multi-table and multi-field rule template, you can add a maximum of five database/table parameters, 20 field parameters, and five input parameters.

- **Output Description**: Enter the description of each column in the SQL statement execution result, which corresponds to the output defined by the relationship in sequence. Column descriptions are separated by commas (,).

  For example, if the relationship is set to **select max (${Column1}), min(${Column2}) from ${Schema_Table1}**, the output result is **Maximum value,Minimum value**.

- **Abnormal Table Template**: You need to enter a complete SQL statement to specify the abnormal data to be exported. You can click **Add Database/Table Parameter** to generate **${Schema_Table1}** which indicates the name of the anomaly table. You can click **Add Field Parameter** to generate **${Column1}** which indicates a field in the anomaly table. You can click **Add Output Parameter** to generate **${Output_Columns}** which indicates the abnormal data to be output from the anomaly table. The system can verify the semantics of the abnormal table template.

  NOTE

  If you enable **Multi-table and multi-field rule**, **Abnormal Table Template** is unavailable.

For example, in a table involving amount, the **is_test** field identifies whether a piece of data is test data (**0** indicates formal data and **1** indicates test data). If you want to calculate the minimum, maximum, average, and total amount of formal data, you can define the custom template as follows:

- **Dimension**: Select **Accuracy**.
- **Directory**: Retain the default value **/All/**.
- **Description**: Enter **Calculate the minimum, maximum, average, and total amount of formal data**.
- **Relationship**: Enter the following SQL statement to calculate the minimum, maximum, average, and total amount of formal data. **${Schema_Table1}** indicates the table selected in the quality job, and **${Column1}** indicates the field selected in **${Schema_Table1}**.

```
select
  min(${Column1}),
  max(${Column1}),
  ROUND(avg(${Column1}),2),
  sum(${Column1})
```

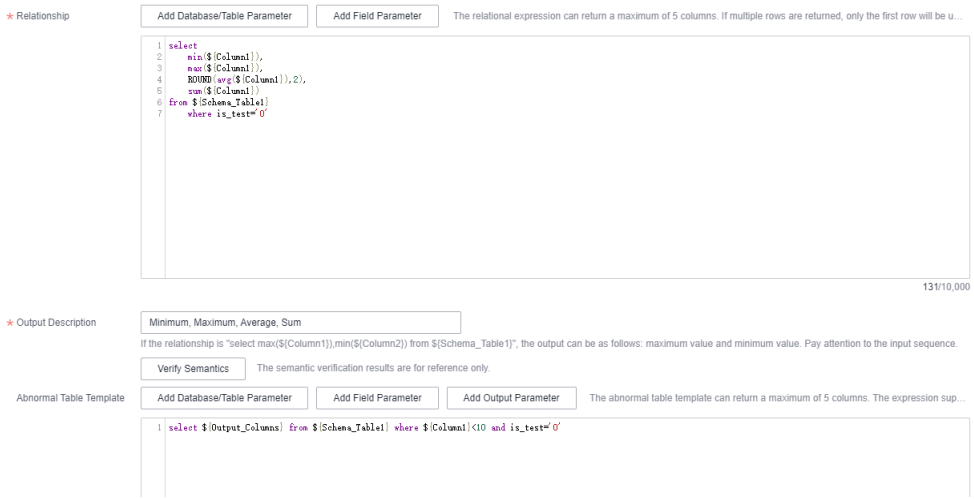


```
from ${Schema_Table1}
  where is_test='0'
```

- **Output Description**: Enter **minimum, maximum, average, and total amount**.
- **Abnormal Table Template**: Enter the following SQL statement to export the **${Output_Columns}** columns in which the amount is less than 10 as abnormal table data. **${Output_Columns}** indicates the field selected for the abnormal table parameter in the quality job.

```
select ${Output_Columns} from ${Schema_Table1} where ${Column1}<10 and is_test='0'
```

Figure 7-11 Key parameters for a custom rule template



Step 4 After you click **Yes**, the system publishes the rule template by default. The default version is V1.0.

----End

Editing Rule Templates

NOTE

Developers cannot randomly modify custom rule templates because they may be used by many users. To modify custom rule templates, contact the administrator.

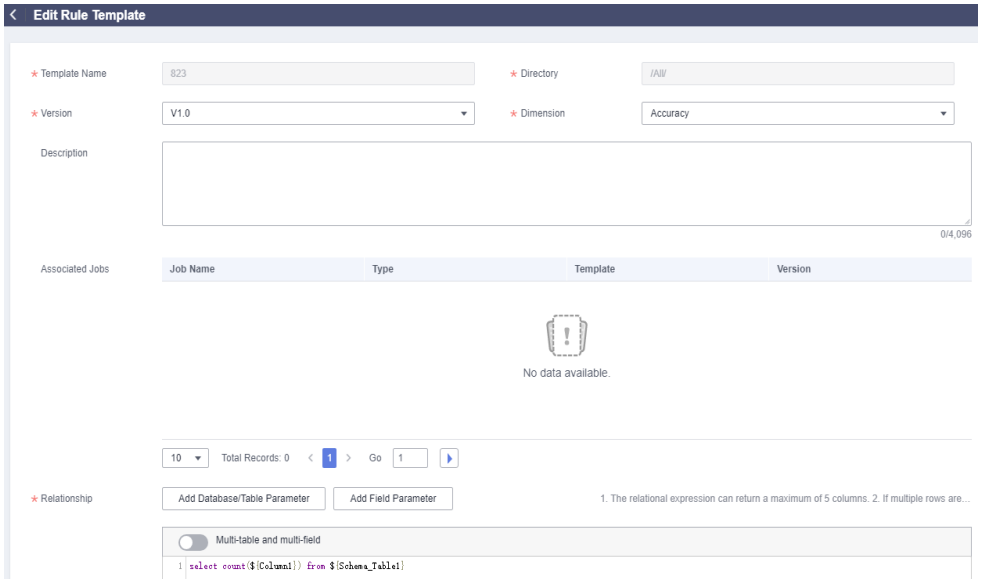
You can edit and publish rule templates. You can take a historical version offline and migrate the jobs associated with the historical version to be taken offline to the new version. The operations are as follows:

NOTE

The page for editing a rule template contains parameters **Version** and **Associated Jobs**.

Step 1 On the DataArts Quality console, choose **Quality Monitoring** > **Rule Templates** in the left navigation pane. Locate the target rule template in the displayed list and click **Edit** in the **Operation** column.

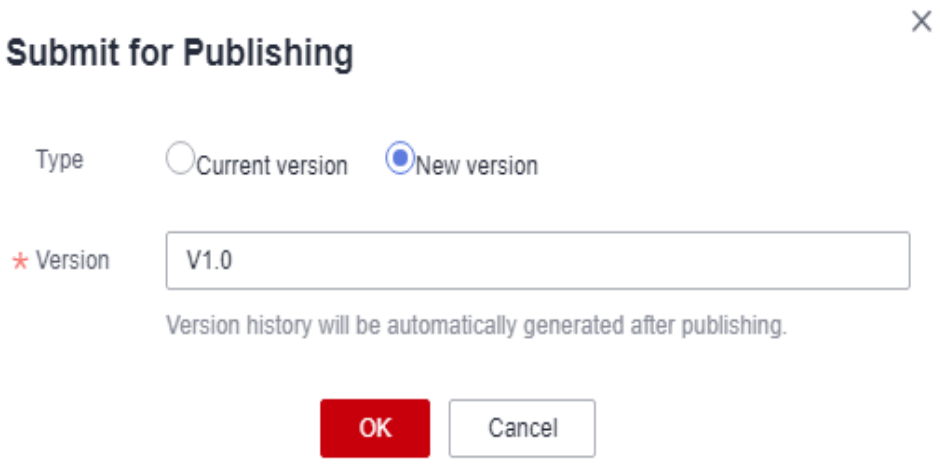
Figure 7-12 Editing a rule template



Step 2 Dimensions and output description can be modified, and relationships can be redefined.

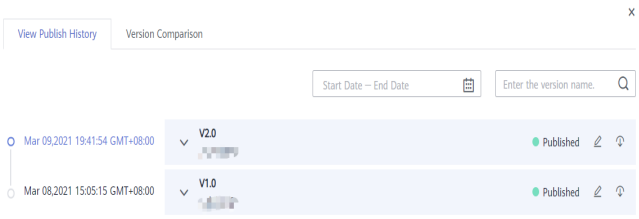
Step 3 Click **Publish**. In the displayed dialog box, select the version type, set the version name, and click **OK**.

Figure 7-13 Publishing a new version



Step 4 After the rule template is submitted for publishing, you can click **View Publish History** in the **Operation** column. You can view the publish history, change the version, and suspend the version.

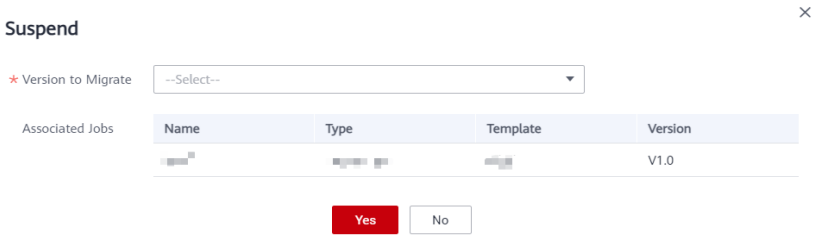
Figure 7-14 Publish History page



Step 5 To suspend a historical version, click **Suspend** on the right of the historical version.

- If the version is not associated with any job, click **OK** to suspend it.
- If the version has associated jobs, select a new version, associate the jobs with the new version, and click **OK**.

Figure 7-15 Migrating and suspending a version



Step 6 On the **Version Comparison** tab page, you can compare the versions to see their differences.

Figure 7-16 Version comparison

----End

Exporting Rule Templates

To export custom rule templates, perform the following steps (you can export a maximum of 200 rule templates at a time):

Step 1 In the left navigation pane, choose **Quality Monitoring** > **Rule Templates**, and select the templates to export in the right pane.

Step 2 Click **Export**. The **Export Rule Template** dialog box is displayed.

Step 3 Click **Export** to switch to the **Export Records** tab.

Step 4 In the list of exported files, locate an exported template and click **Download** in the **Operation** column to download the Excel file of the rule template to the local PC.

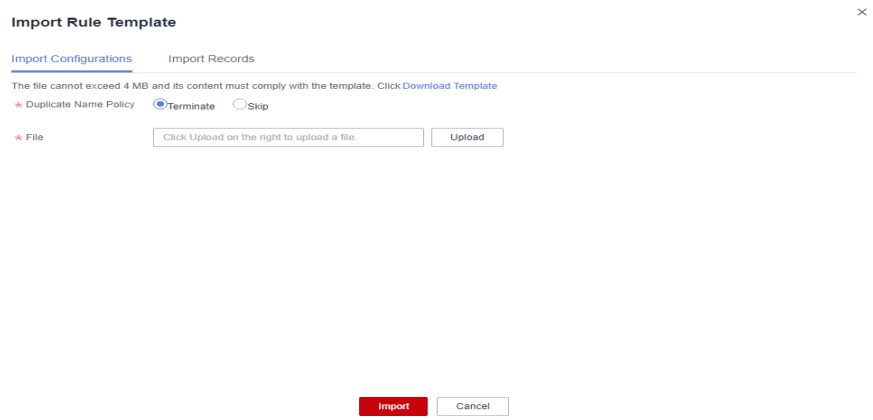
----End

Importing Rule Templates

You can import a file containing a maximum of 4 MB data.

Step 1 In the left navigation pane, choose **Quality Monitoring** > **Rule Templates**. In the right pane, click **Import**.

Figure 7-17 Importing rule templates



Step 2 On the **Import Configurations** tab page, set **Duplicate Name Policy**.

- **Terminate**: If template names repeat, all templates will fail to be imported.
- **Skip**: If template names repeat, the templates will still be imported.

Step 3 Click **Upload** and select the prepared data file.

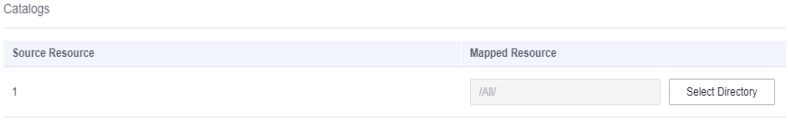
NOTE

Edit the data file using either of the following methods:

- (Recommended) Click **Export** to export data and import the data to the system directly or import it after modification.
- Click **Download Template**, fill in the template with the data to import, and import the data to the system.

Step 4 Configure the mapped resource for the catalog and select the directory where rule template has been imported. If you do not configure the resource mapping, the original mapping is used by default.

Figure 7-18 Configuring the resource mapping



Step 5 Click **Import** to import the Excel template to the system.

Step 6 Click the **Import Records** tab to view the import records.

----End

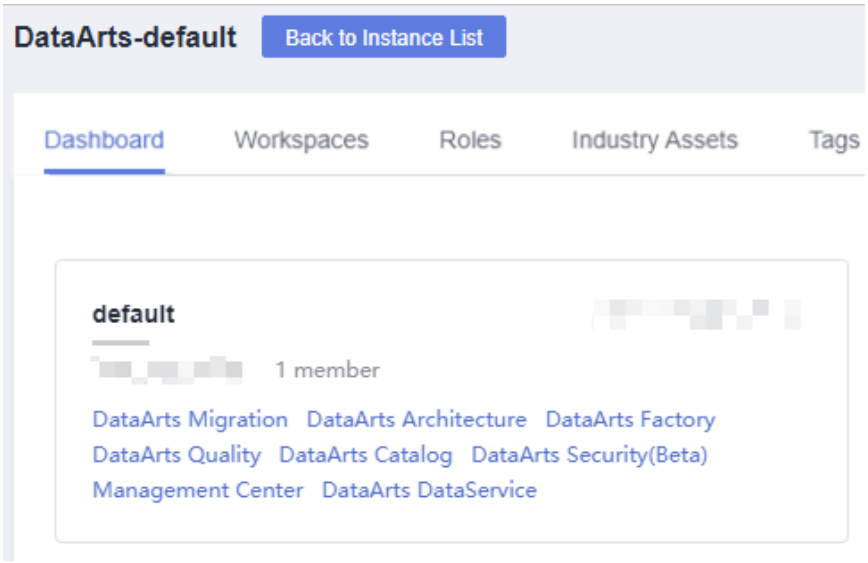
7.2.3 Creating Quality Jobs

You can create quality jobs to apply the created rules to existing tables.

Procedure

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Quality**.

   Figure 7-19 DataArts Quality

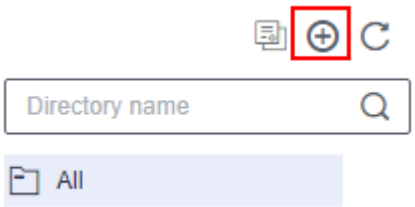


2. (Optional) In the left navigation pane, choose **Quality Monitoring** > **Quality Jobs** and create a directory. If a directory exists, you do not need to create one. Note that rule templates, quality jobs, and comparison job are in the same directory.

   Currently, you can create a directory using either of the following methods:

   Click ⊕ and enter a directory name in the displayed dialog box. In this way, you can create a maximum of seven layers of directories.

   Figure 7-20 Creating a directory

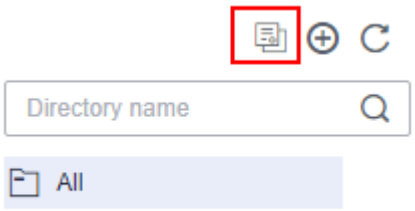


You can also click to synchronize the **subjects in DataArts Architecture** as directories. (Only published subjects can be synchronized.) The synchronized directories are consistent with the published subjects in DataArts Architecture and are displayed by layer, such as L1 and L2.

NOTE

1. The directories you create are not affected by the synchronization. If the name of a created directory conflicts with that of a subject:
   - If they conflict during the first synchronization, a subject layer (such as L1 and L2) is added to the name of the directory.
   - If they conflict after the subject is modified, the synchronization fails.
2. Changes to subjects or subject layers in DataArts Architecture cannot be automatically synchronized. You must click again to synchronize them.
   If a subject or subject layer in DataArts Architecture is deleted and synchronized to DataArts Quality, the corresponding directory will not be deleted. Instead, the subject attributes will be deleted from the directory.
3. After the synchronization is complete, the system automatically displays the synchronization result details. You can view the names of the subjects that fail to be synchronized.

Figure 7-21 Synchronizing subjects from DataArts Architecture



3. On the **Quality Jobs** page, click **Create**. In the dialog box displayed, set the parameters based on **Table 7-11**.

Table 7-11 Quality job parameters

Parameter	Description
*Job Name	Quality job name
Description	Information to better identify the quality job. It cannot exceed 1,024 characters.
*Directory	Directory for storing the quality job. You can select a created directory. For details about how to create a directory, see **(Optional) Creating a Directory**.
*Job Level	The options are **Warning**, **Minor**, **Major**, and **Critical**. The job level determines the template for sending notification messages.
Issue Handler	Handler of the issues detected by the quality job

4. Click **Next** to go to the **Define Rule** page, on which each rule card corresponds to a subjob. Click on the rule card and configure it based on Table 7-12. You can also add more quality rules and click **Next** to apply them to a created database or table.

Figure 7-22 Configuring rules for a quality job

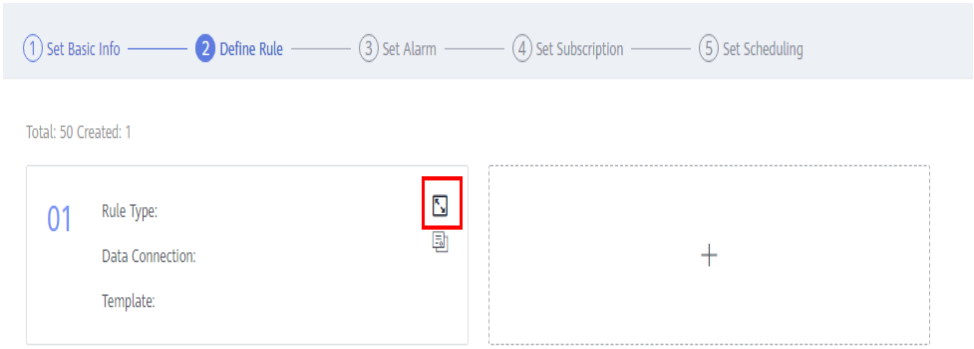


Table 7-12 Parameters for configuring a rule

Parameter	Sub-parameter	Description
Basic Information	Subjob Name	In the job execution result, each rule corresponds to a subjob. You are advised to set the subjob information so that you can view the job execution result and locate faults through logs more easily.
	Description	Information to better identify the subjob It cannot exceed 1,024 characters.
Object	Rule Type	Database rule, table rule, field rule, cross-field rule, multi-table and multi-field rule, or custom rule configured for specific fields in a table.

Parameter	Sub-parameter	Description

	Data Connection	Source and destination objects support the following data connection types: DWS, MRS Hive, MRS Spark, DLI, RDS (MySQL and PostgreSQL), Oracle, MRS Spark (Hudi), and MRS ClickHouse.
		Select a created data connection from the drop-down list box.
		NOTE
		• Rules are based on data connections. Therefore, you must create data connections in **Management Center** before creating data quality rules.
		• For MRS Hive connected through a proxy, select the MRS API mode or proxy mode.
		  • MRS API mode: An MRS API is used for submission. By default, historical jobs are submitted through MRS APIs. You are advised keep the default settings when editing the job.
		  • Proxy mode: A username and a password are used for submission. You are advised to select this mode for new jobs to prevent job submission failures caused by permission issues.
		• The strict mode of the MRS Hive component is not supported.

	Database	Select the database to which the configured data quality rules are applied. **NOTE** • The database is tailored to the created data connection. • When **Rule Type** is set to **Database rule**, set the data object to the corresponding database.
	Data Table	Select the table to which the configured data quality rules apply. **NOTE** • The table is closely related to the database. • When **Rule Type** is set to **Table rule**, set the data object to the corresponding table.
	SQL	This parameter is mandatory if you select **Custom rule** for **Rule Type**. Enter a complete SQL statement to define how to monitor the quality of data objects.
	Failure Policy	Select **Ignore rule errors** as required.

Parameter	Sub-parameter	Description
	Select Fields	This parameter is mandatory if you select **Field rule** for **Rule Type**. Select a field in the corresponding data table. **NOTE** Fields names containing only one letter (such as **a**, **b**, **c**, and **d**) cannot be verified.
	Data Object	This parameter is mandatory if you select **Cross-field rule** for **Rule Type**. Select a reference data field. When you select a table name, the search box is case sensitive.
	Reference Data Object	This parameter is mandatory if you select **Cross-field rule** for **Rule Type**. Select a reference data field. When you select a table name, the search box is case sensitive.
	Dimension	This parameter is mandatory if you select **Custom rule** for **Rule Type**. It associates the custom rule with one of the six quality attributes, including completeness, validity, timeliness, consistency, accuracy, and uniqueness.
Compute Engine	Cluster Name	Select the engine for running the quality job. This parameter is valid only for DLI data connections.
Rule Template	Template	Select a system or custom rule template. **NOTE** The template type is closely related to the rule type. For details, see **Table 7-10**. In addition to system rule templates, you can select the custom rule template created in **Creating Rule Templates**. If **Rule Type** is set to **Field rule** and **Rule Template** is set to **Regular expression verification** or **Regular expression verification ignore null**, the regular expression rule can contain a maximum of 1,024 characters.
	Version	This parameter is required only when you select a custom rule template. Select the version of the published custom rule template.
	Scoring Weight	Set the weight for the rule based on the field level. The value is an integer from 1 to 9. The default value is **5**.

Parameter	Sub-parameter	Description
Object Scope	Scanning Scope	You can select **All** or **Partial**. The default value is **All**. If you want only part of data to be computed or quality jobs to be executed periodically based on a timestamp, you can set a WHERE condition for scanning. You can transfer environment variables to data quality jobs. If rules can be configured for multiple tables, the data range of each table can be set independently. If both **Data Object** and **Reference Data Object** are set, you need to configure the data to scan in the scanning scope.

	WHERE Clause	Enter a WHERE clause. The system will scan the data that matches the clause.
		For example, if you want to filter out the data for which the value range of the **age** field is (18, 60], enter the following WHERE clause:
		`age > 18 and age <= 60`
		You can also enter a dynamic SQL expression. For example, if you want to filter out the data generated 24 hours ago based on the **time** field, enter the following WHERE clause:
		`time >= (date_trunc('hour', now()) - interval '24 h') and time <= (date_trunc('hour', now()))`
		DataArts Quality allows you to transfer parameters. You can enter a condition expression to transfer environment variables. The following is an example:
		`p_date=${target_date}`
		You can also transfer parameters from DataArts Factory to DataArts Quality. DataArts Quality can also proactively obtain parameters from DataArts Factory. This is supported for both system and custom rule templates.

	Parameter Value	This parameter is required when you select **Partial** for **Scanning Scope**. Enter the default values of the parameters in the where clause. After DataArts Factory transfers parameters to DataArts Quality and the job is executed, you can click **View SQL** to view the parameters and their values transferred by DataArts Factory.

Parameter	Sub-parameter	Description
Alarm Condition	Alarm Expression	Set this parameter if you want to set an alarm condition for the current rule. If you want to use the logical operations of multiple rules to set a unified alarm condition expression, you do not need to set this parameter. Instead, you can set it on the next **Set Alarm** page. After the alarm conditions are configured, the system determines whether to generate an alarm based on the value of **Parameter** and the alarm condition. Apart from a single alarm expression, you can also use more complex alarm conditions consisting of logical operators. The alarm expression supports the following logical operators, which can be enclosed by "(" and ")". ● **+**: addition ● **-**: subtraction ● ***:** multiplying ● **/**: division ● **==**: equal to ● **!=**: not equal to ● **>**: greater than ● **<**: less than ● **>=**: greater than or equal to ● **<=**: less than or equal to ● **!**: non ● **||**: or ● **&&**: and For example, if **Rule Template** is set to **Null value**, you can set this parameter as follows: ● If you want an alarm to be generated when the number of rows with a null value is greater than 10, enter **${1}>10** (**${1}** is the number of rows with a null value). ● If you want an alarm to be generated when the ratio of fields with a null value is greater than 80%, enter **${3}>0.8** (**${3}** is the ratio of fields with a null value). ● If you want an alarm to be generated when the number of rows with a null value is greater than 10 or the ratio of fields with a null value is greater than 80%, enter **(${1}>10)||(${3}>0.8)** (**${1}** is the number of rows with a null value, **${3}** is the ratio of fields with a null value, and **||**

Parameter	Sub-parameter	Description
		indicates that an alarm will be generated if either of the conditions is met).
	Parameter	The value of this parameter is obtained from the output of the rule template. If you can click a parameter, the expression of the parameter is displayed in **Alarm Expression**. For example, if **Template** is set to **Null value**, **${1}** is displayed in **Alarm Expression** when you click alarm parameter **Null Value Rows**.
	Logical Operator	This parameter is optional. You can perform logical operations on the result of an alarm expression to generate more complex alarm conditions. You can move the cursor between two alarm expressions in **Alarm Expression** and click one of the following operators to insert them. You can also manually enter an operator. The current expression supports the following logical operators which can be enclosed by brackets (). ● **+**: addition ● **-**: subtraction ● ***:** multiplying ● **/**: division ● **==**: equal to ● **!=**: not equal to ● **>**: greater than ● **<**: less than ● **>=**: greater than or equal to ● **<=**: less than or equal to ● **!**: non ● **||**: or ● **&&**: and For example, if **Template** is set to **Null value** and you want an alarm to be generated when the number of rows with a null value is greater than 10 or the ratio of fields with a null value is greater than 80%, enter **(${1}>10)||(${3}>0.8)** for **Alarm Expression** (**${1}** is the number of rows with a null value, **${3}** is the ratio of fields with a null value, and **||** indicates that an alarm will be generated if either of the conditions is met).

Parameter	Sub-parameter	Description
	Score Quality	This parameter is mandatory if you select **Custom rule** for **Rule Type**.
	Generate Anomaly Data	Enable **Generate Anomaly Data** and click **Select** next to **Anomaly Table** to store the anomaly data that does not comply with the preset rules. **NOTE** • For a field rule, the average value, total value, maximum value, and minimum value of a field in the field-level rule template cannot be used to generate anomaly data. • If periodic scheduling or re-execution is configured for a quality job, abnormal data detected in each instance scan is inserted into the anomaly table. You are advised to periodically delete the data in the anomaly table to reduce cost and ensure good performance.
	Anomaly Table	Select a database table. You can configure the prefix and suffix of the output table name. **NOTE** When you set an anomaly table, the system adds suffix **err** to the table name by default.
	Output Settings	• **Output Rule Settings**: If you select this option, the quality job settings will show up in the anomaly tables so that you can view the anomaly data sources with ease. • **Output null**: If you select this option, and the preset rules are not complied, the null value will show up in anomaly tables.
	Anomaly Data Amount	You can choose to export all anomaly data or the specified amount of anomaly data.
	Anomaly Table SQL	This parameter is mandatory if you select **Custom rule** for **Rule Type**. You need to enter a complete SQL statement to specify the abnormal data to be exported.
	View Duplicate Rules	Click it to view the following duplicate rules: • Determine the rule repetition based on tables and fields. • View the related sub-rules and quality jobs that already exist.

5. Click **Next** and set alarm information. If you have configured an alarm expression in the previous step, the configured expression is automatically

displayed. If there are two or more sub-rules, you can use either of the following methods to configure alarms:

a. Use the alarm conditions of sub-rules to report alarms.
b. Perform mathematical and logical operations on the alarm parameter values to generate a universal alarm expression to specify whether to report alarms for jobs.

The alarm expression supports the following logical operators, which can be enclosed by "(" and ")".

- **+**: addition
- **-**: subtraction
- *****: multiplying
- **/**: division
- **==**: equal to
- **!=**: not equal to
- **>**: greater than
- **<**: less than
- **>=**: greater than or equal to
- **<=**: less than or equal to
- **!**: non
- **||**: or
- **&&**: and

6. Click **Next** and set the subscription information. If the SMN notification is required, enable **Notification**, and set **Notification Type** and **Topic**.

   NOTE

   After notification is enabled, a notification is sent for all the subjobs of the configured notification type.

   If you enable alarming, you do not need to set the notifications for failures. Alarms will be automatically reported if a task fails.

   Currently, only SMS and email are available for subscribing to topics.

   You can select **Alarm triggered** or **Run successfully** for **Notification Type**.

   If you enable **Notification Policy**, you can set the condition for sending an alarm notification, that is, the number of consecutive alarms within a certain period of time (minutes). The period ranges from 1 minute to 360 minutes, and the number of consecutive times ranges from 1 to 10.

7. Click **Next** to go to the page where you can select a scheduling mode. Currently, **Once** and **On schedule** are supported. Set parameters for scheduling periodically by referring to **Table 7-13**. Click **Submit**.

NOTE

1. If **Once** is selected, a manual task instance is generated. A manual task has no dependency on scheduling and must be manually triggered.
2. If **On schedule** is selected, a periodic instance is generated. A periodic instance is an instance snapshot that is automatically scheduled when a periodic task reaches the scheduled execution time.
3. When a periodic task is scheduled once, an instance workflow is generated. You can perform routine O&M on scheduled instance tasks, such as viewing the running status, stopping and rerunning the scheduled tasks.
4. Only MRS clusters that support job submission through an agency support periodic scheduling of quality jobs. MRS clusters that support job submission through an agency are as follows:
   - MRS non-security cluster
   - MRS security cluster whose version is later than 2.1.0, and that has MRS 2.1.0.1 or later installed

Table 7-13 Parameters

Parameter	Description
Effective	Effective date of a scheduling task.
Cycle	The frequency at which a scheduling task is executed. Related parameters are: • Minutes • Hours • Days • Weeks **NOTE** • If **Cycle** is set to **Minutes** or **Hours**, set the start time, end time, and interval for the scheduling task. Currently, the start time is in minute for stagger scheduling. • If **Cycle** is set to **Days**, set a specified time when the scheduling task is enabled every day. • If **Cycle** is set to **Weeks**, set **Scheduling Time** and **Start from** for the scheduling task, that is, *XX* o'clock *XX* minutes on *XXX* every week.

After a quality job is created, you can view it in the job list. You can also filter jobs by job name, creator, owner, table name, and time range. The system supports fuzzy search.

After a quality job is created, you can edit, delete, run, start scheduling, and stop scheduling it.

NOTE

You cannot start scheduling a one-off quality job.

Running a Quality Job

To run a quality job, perform the following operations:

Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **Quality Jobs**. In the right pane, locate a quality job.

Step 2 Click **Run** in the **Operation** column.

Step 3 In enterprise mode, select the development environment or production environment.

Step 4 Click **OK**.

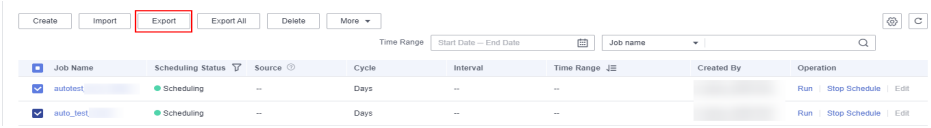
----End

Exporting Quality Jobs

You can export a maximum of 200 quality jobs. Each cell of the exported file can contain a maximum of 65,534 characters.

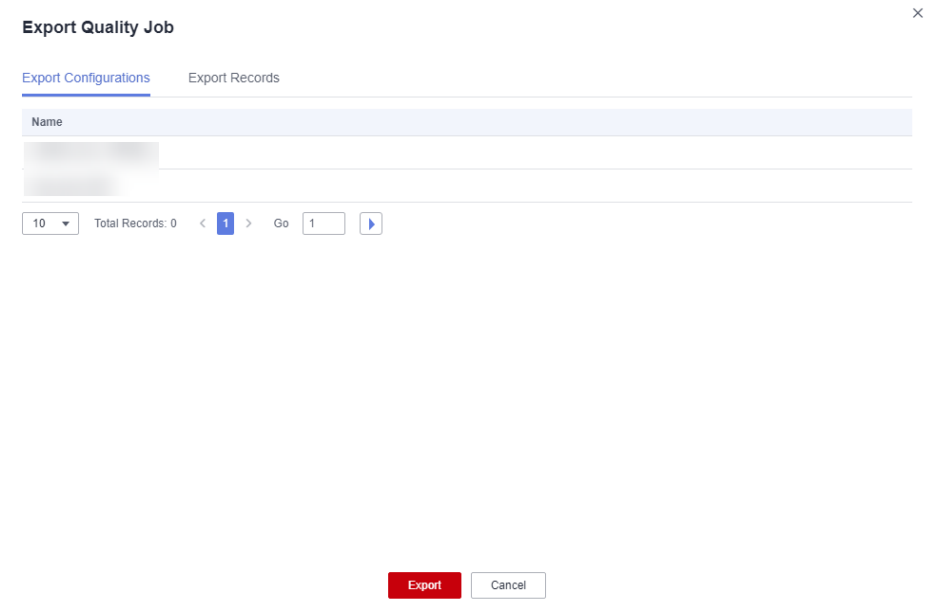
Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **Quality Jobs**. In the right pane, select the quality jobs to export.

Figure 7-23 Export



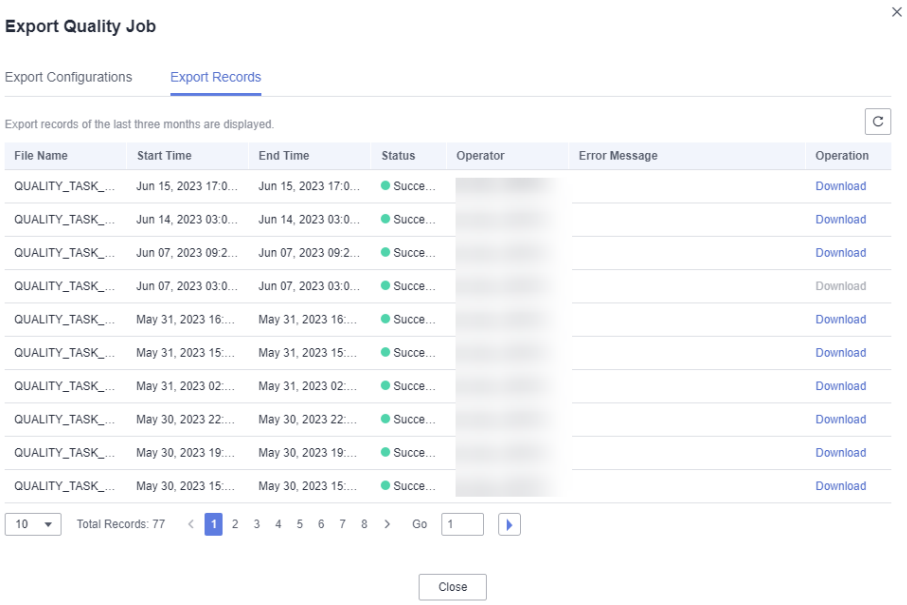
Step 2 Click **Export**. The **Export Quality Job** dialog box is displayed.

Figure 7-24 Exporting Quality Jobs



Step 3 Click the **Export Records** tab to view the export result.

Figure 7-25 Export Records



Step 4 In the list of exported files, locate an exported quality job and click **Download** in the **Operation** column to download the Excel file of the job to the local PC.

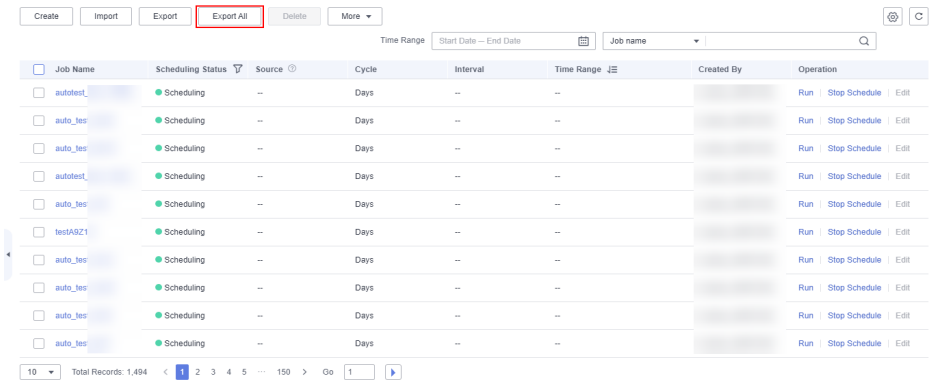
----End

Exporting All Quality Jobs

To export all quality jobs, perform the following operations: Each cell of the exported file can contain a maximum of 65,534 characters.

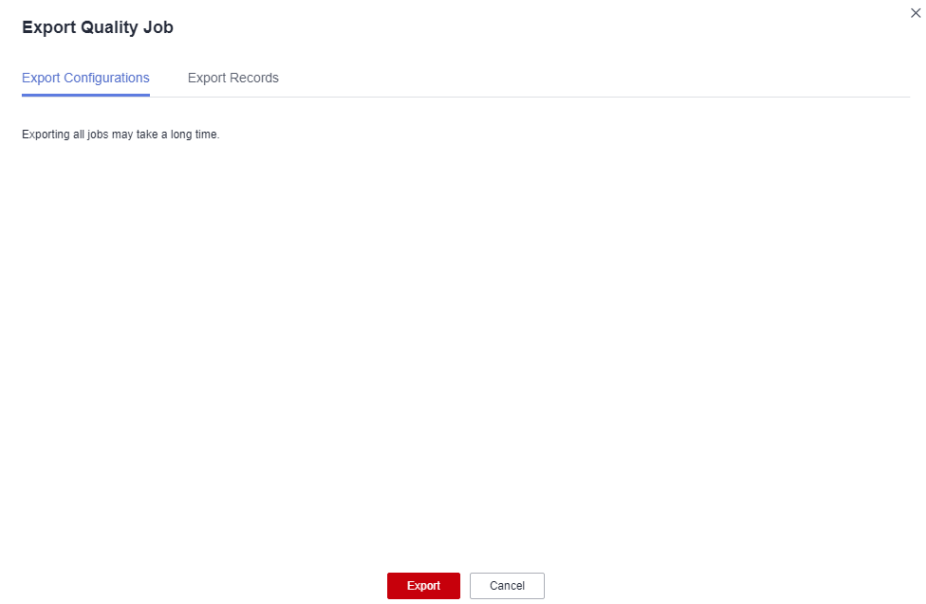
Step 1 Choose **Quality Monitoring** > **Quality Jobs** and click **Export All**.

Figure 7-26 Export All



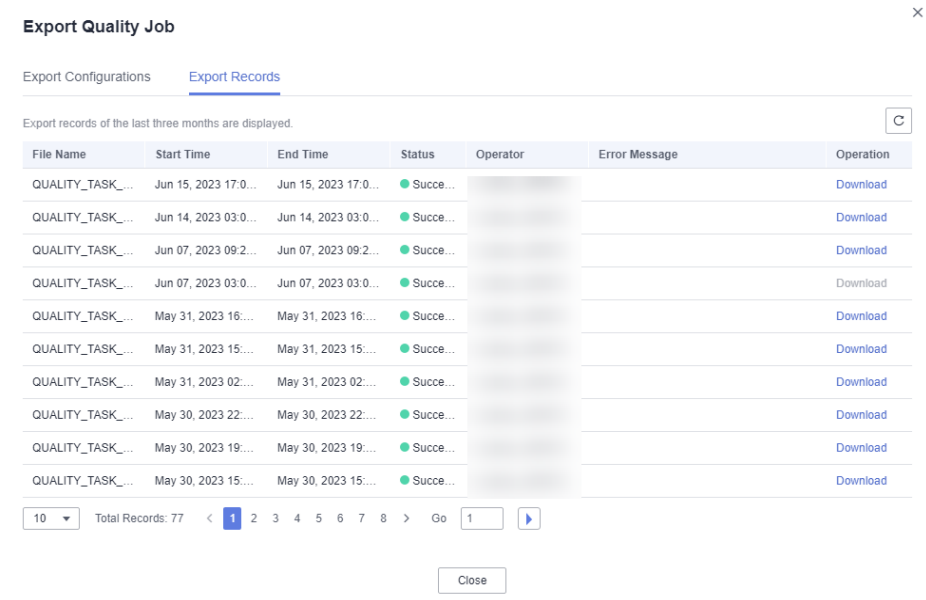
Step 2 In the displayed **Export Quality Job** dialog box, click **Export**.

Figure 7-27 Exporting all quality jobs



Step 3 Click the **Export Records** tab to view the export result.

Figure 7-28 Export Records



Step 4 In the list of exported files, locate an exported quality job and click **Download** in the **Operation** column to download the Excel file of the job to the local PC.

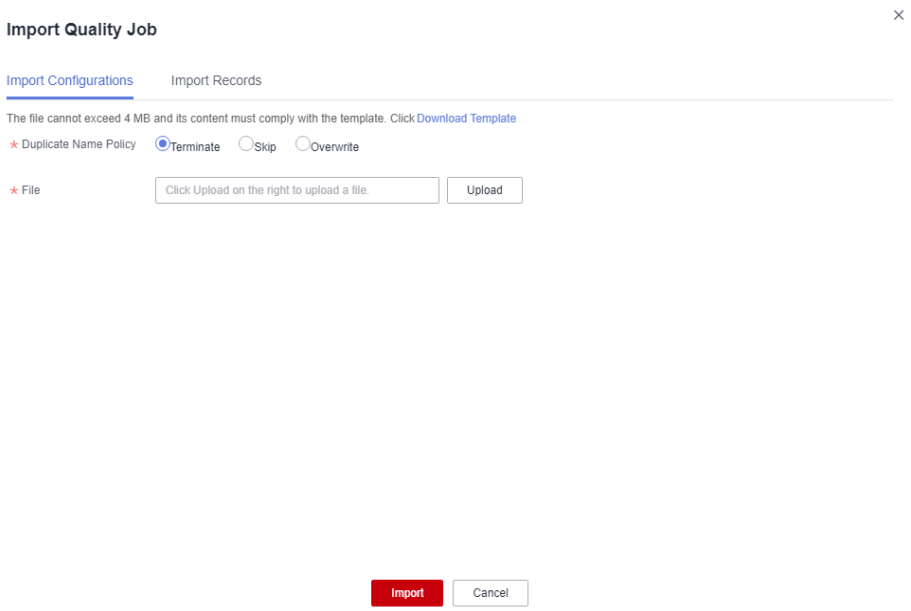
----End

Importing Quality Jobs

You can import a file containing a maximum of 4 MB data. Each cell of the file to be imported can contain a maximum of 65,534 characters.

Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **Quality Jobs**. In the right pane, click **Import**. The **Import Quality Job** dialog box is displayed.

Figure 7-29 Importing quality jobs



Step 2 On the **Import Configurations** tab page, set **Duplicate Name Policy**.

- **Terminate**: If quality job names repeat, all quality jobs will fail to be imported.
- **Skip**: If quality job names repeat, the quality jobs will still be imported.
- **Overwrite**: If quality job names repeat, new jobs will replace existing ones with the same names.

  NOTE

  If you select **Overwrite**, stop job scheduling before uploading a file. Otherwise, the upload will fail.

Step 3 Click **Upload** and select the prepared data file.

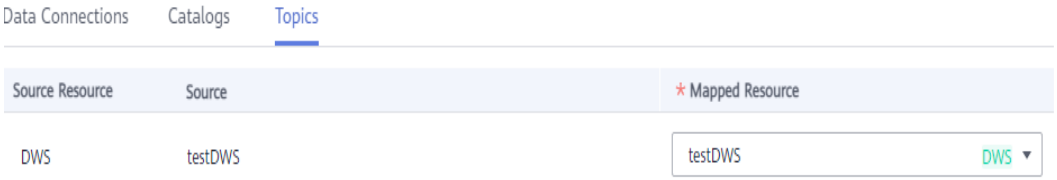
NOTE

Edit the data file using either of the following methods:

- (Recommended) Click **Export** to export data and import the data to the system directly or import it after modification.
- Click **Download Template**, fill in the template with the data to import, and import the data to the system.

Step 4 Configure resource mapping for the data connection, cluster, catalog, and topic. If you do not configure the resource mapping, the original mapping is used by default.

Figure 7-30 Configuring the resource mapping



- **Data Connection**: Select the type of the imported data connection.
- **Cluster**: If the data connection type is DLI, select the corresponding queue.
- **Directory**: Select the directory where the imported quality job is stored.
- **Topic**: If SMN is configured, you need to select a topic.

Step 5 Click **Import** to import the Excel template to the system.

Step 6 Click the **Import Records** tab to view the import records.

----End

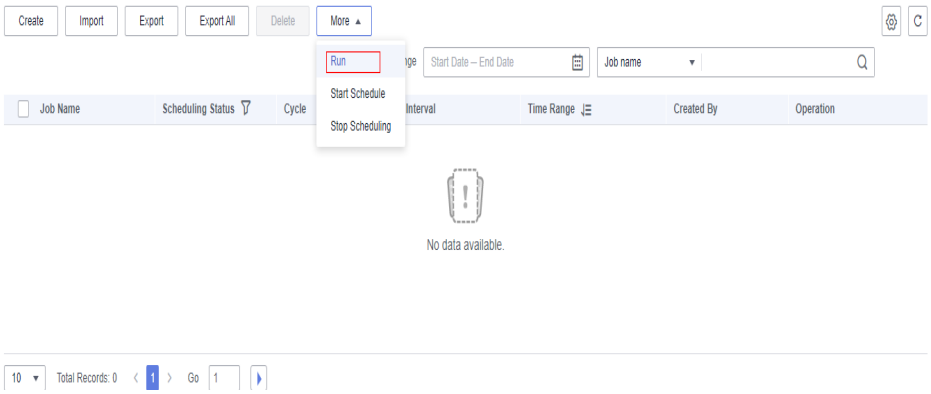
Running Quality Jobs

You can run a maximum of 200 quality jobs.

Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **Quality Jobs**. In the right pane, select the quality jobs you want to run.

Step 2 Above the job list, click **More** and select **Run** to run the selected quality jobs.

Figure 7-31 Running jobs



Step 3 In enterprise mode, select the development environment or production environment.

Step 4 Click **OK**.

----End

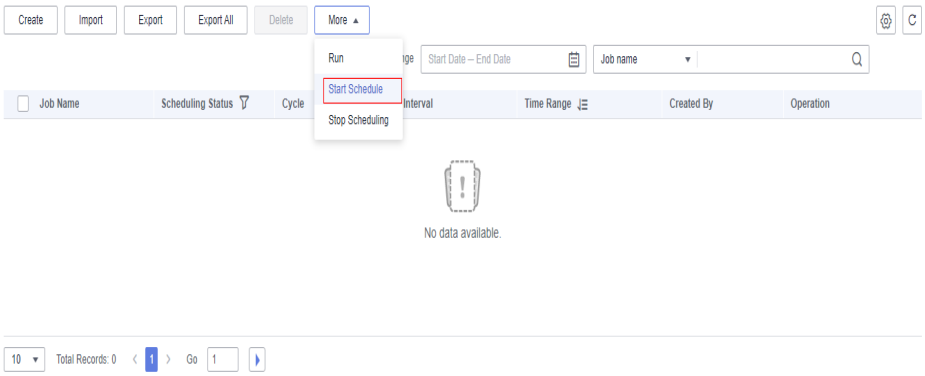
Scheduling Quality Jobs

You can schedule a maximum of 200 quality jobs at a time.

Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **Quality Jobs**. In the right pane, select the quality jobs you want to schedule.

Step 2 Above the job list, click **More** and select **Start Schedule** to schedule the selected quality jobs.

Figure 7-32 Scheduling jobs



----End

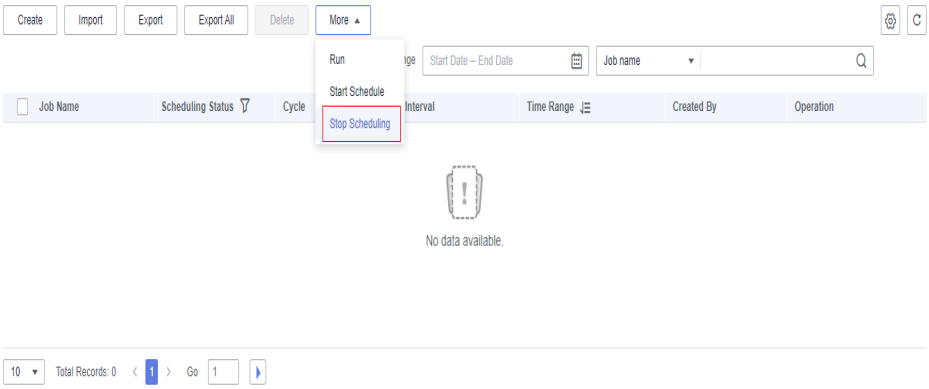
Stopping Scheduling Quality Jobs

You can stop scheduling a maximum of 200 quality jobs at a time.

Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **Quality Jobs**. In the right pane, select the quality jobs you want to stop scheduling.

Step 2 Above the job list, click **More** and select **Stop Scheduling** to stop scheduling the selected quality jobs.

Figure 7-33 Stopping scheduling jobs



----End

Stopping Quality Jobs

You can stop a maximum of 200 quality jobs at a time.

Only quality jobs in **Running** state can be stopped.

Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **O&M**. In the right pane, select the quality jobs you want to stop.

Step 2 Click **Stop**. In the displayed **Stop Instance** dialog box, confirm the instances to stop and click **Yes**.

Figure 7-34 Stopping instances

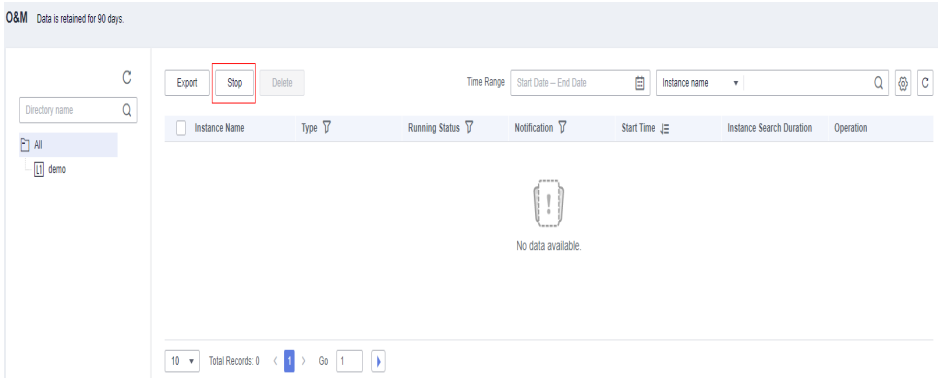
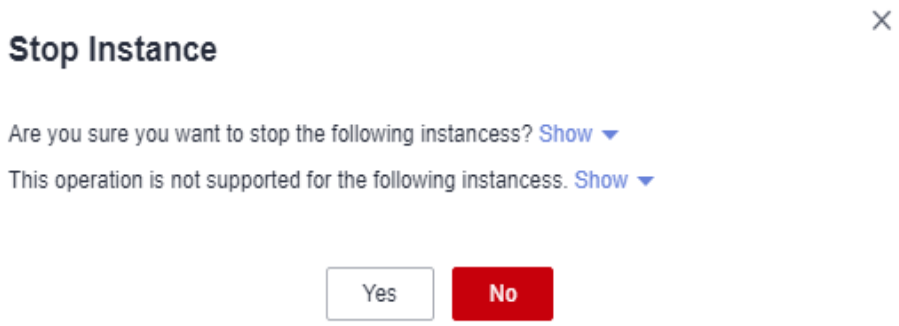


Figure 7-35 Stopping instances



----End

7.2.4 Creating a Comparison Job

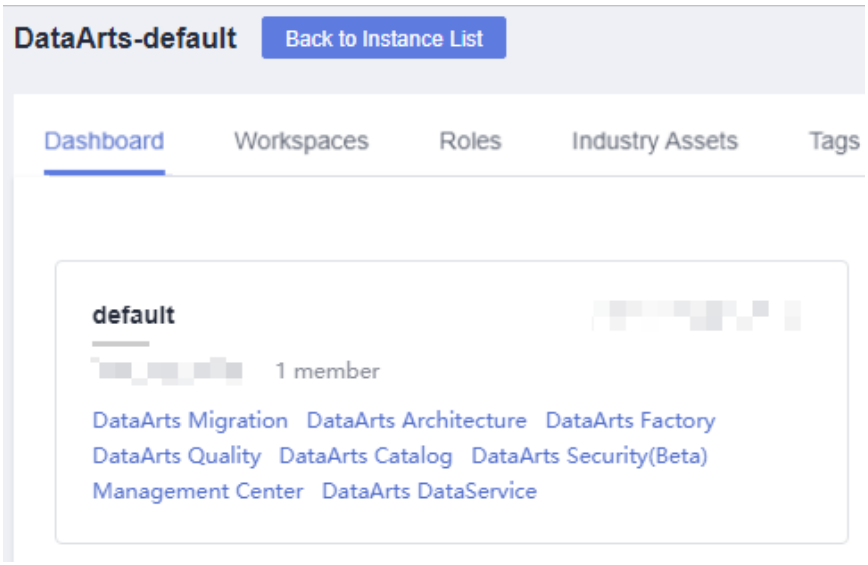
Data comparison is critical to ensure data consistency in data development and migration. The cross-source data comparison capability is the key to checking consistency of the data before and after migration or processing.

Comparison jobs in Quality Monitoring support cross-source data comparison. You can apply created rules to two tables for quality monitoring and output the comparison result.

Creating a Job

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Quality**.

   Figure 7-36 DataArts Quality

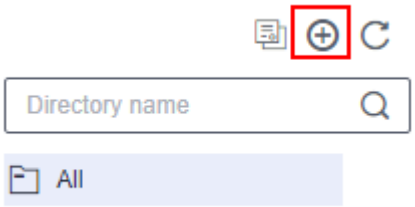
   

2. (Optional) In the left navigation pane, choose **Quality Monitoring** > **Comparison Jobs** and create a directory. If a directory exists, you do not need to create one. Note that rule templates, quality jobs, and comparison job are in the same directory.

   Currently, you can create a directory using either of the following methods:

   Click ⊕ and enter a directory name in the displayed dialog box. In this way, you can create a maximum of seven layers of directories.

   Figure 7-37 Creating a directory

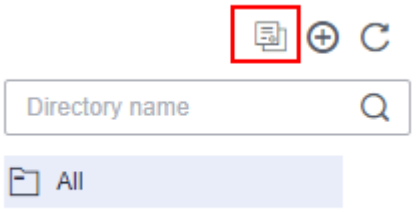
   You can also click to synchronize the **subjects in DataArts Architecture** as directories. (Only published subjects can be synchronized.) The synchronized directories are consistent with the published subjects in DataArts Architecture and are displayed by layer, such as L1 and L2.

NOTE

1. The directories you create are not affected by the synchronization. If the name of a created directory conflicts with that of a subject:
   - If they conflict during the first synchronization, a subject layer (such as L1 and L2) is added to the name of the directory.
   - If they conflict after the subject is modified, the synchronization fails.
2. Changes to subjects or subject layers in DataArts Architecture cannot be automatically synchronized. You must click again to synchronize them.

   If a subject or subject layer in DataArts Architecture is deleted and synchronized to DataArts Quality, the corresponding directory will not be deleted. Instead, the subject attributes will be deleted from the directory.
3. After the synchronization is complete, the system automatically displays the synchronization result details. You can view the names of the subjects that fail to be synchronized.

Figure 7-38 Synchronizing subjects from DataArts Architecture



3. On the **Quality Jobs** page, click **Create**. In the displayed dialog box, set the parameters listed in **Table 7-14**.

Table 7-14 Comparison job parameters

Parameter	Description
Name	Comparison job name
Description	Information to better identify a comparison job. It cannot exceed 1,024 characters.
Directory	The directory for storing the comparison job to create. You can select a created directory. For details about how to create a directory, see **(Optional) Creating a Directory**.
Job Level	The options are **Warning**, **Minor**, **Major**, and **Critical**. The job level determines the template for sending notification messages.

4. Click **Next** to go to the **Define Rule** page. Click on the rule card and configure it based on **Table 7-15**. You can also add comparison rules.

Figure 7-39 Configuring rules for a comparison job

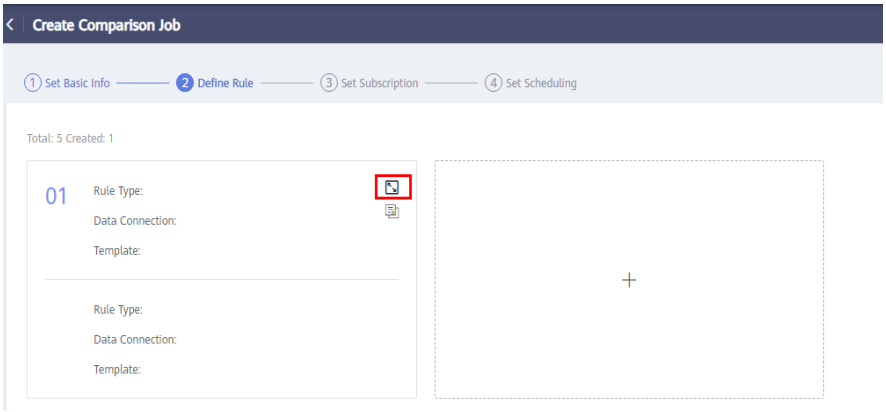


Table 7-15 Parameters for configuring a rule template

Module	Parameter	Description
Basic Information	Subjob Name	In the job execution result, each rule corresponds to a subjob. You are advised to set the subjob information so that you can view the job execution result and locate faults through logs more easily.
	Description	Information to better identify the subjob It can contain a maximum of 1,024 characters.
Object	Rule Type	The options are **Table rule**, **Field rule**, and **Custom rule**. Field-level rules can be used to configure monitoring rules for specific fields in tables. For example, set this parameter to **Table rule**, and set other configuration items on the page to table-level rule configuration items correspondingly. The rule type of the destination object is automatically generated based on that of the source object.

Module	Parameter	Description

	Data Connection	Source and destination objects support the following data connection types: DWS, MRS Hive, MRS Spark, DLI, RDS (MySQL and PostgreSQL), Oracle, MRS Spark (Hudi), and MRS ClickHouse.
		Select a created data connection from the drop-down list box.
		NOTE
		• Rules are based on data connections. Therefore, you must create data connections in **Management Center** before creating data quality rules.
		• For MRS Hive connected through a proxy, select the MRS API mode or proxy mode.
		  • MRS API mode: An MRS API is used for submission. By default, historical jobs are submitted through MRS APIs. You are advised keep the default settings when editing the job.
		  • Proxy mode: A username and a password are used for submission. You are advised to select this mode for new jobs to prevent job submission failures caused by permission issues.
		• The strict mode of the MRS Hive component is not supported.

	Data Object	The data table selected for the source object is compared with the data table of the destination object on the right. Select the table to which the configured comparison rule applies. **NOTE** The table is closely related to the database. The database is tailored to the created data connection.
	SQL	This parameter is mandatory if you select **Custom rule** for **Rule Type**. Enter a complete SQL statement to define how to monitor the quality of data objects.
Compute Engine	Cluster Name	Select the engine for running the comparison job. This parameter is valid only for DLI data connections.

Module	Parameter	Description
Rule Template	Template	This parameter defines how to monitor the quality of data objects. The template name of the source object contains the system rule template and custom rule template. The template name of the destination object is automatically generated based on the rule type of the source object. **NOTE** The template type is closely related to the rule type. For details, see **Table 7-10**. In addition to system rule templates, you can select the custom rule template created in **Creating Rule Templates**. If **Rule Type** is set to **Field rule** and **Rule Template** is set to **Regular expression verification** or **Regular expression verification ignore null**, the regular expression rule can contain a maximum of 1,024 characters.
	Version	This parameter is required only when you select a custom rule template. Select the version of the published custom rule template.
Object Scope	Scanning Scope	You can select **All** or **Partial**. The default value is **All**. If you want only part of data to be computed or comparison jobs to be executed periodically based on a timestamp, you can set a where clause for scanning.

	WHERE Clause	Enter a WHERE clause. The system will scan the data that matches the clause.
		For example, if you want to filter out the data for which the value range of the **age** field is (18, 60], enter the following WHERE clause:
		`age > 18 and age <= 60`
		You can also enter a dynamic SQL expression. For example, if you want to filter out the data generated 24 hours ago based on the **time** field, enter the following WHERE clause:
		`time >= (date_trunc('hour', now()) - interval '24 h') and time <= (date_trunc('hour', now()))`

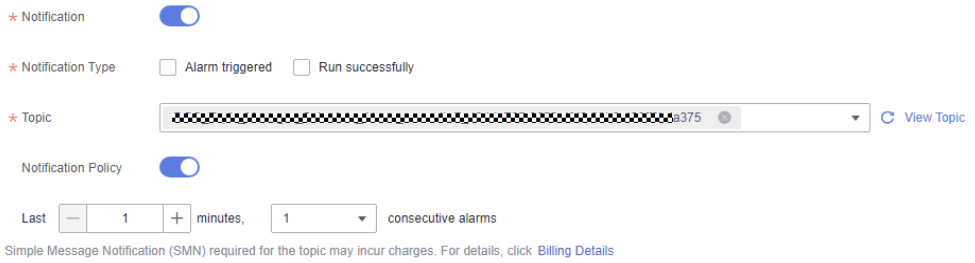
Module	Parameter	Description
Alarm Condition	Alarm Expression	Set this parameter if you want to set an alarm condition for the current rule. After the alarm conditions are configured, the system determines whether to generate an alarm based on the value of **Parameter** and the alarm condition. Apart from a single alarm expression, you can also use more complex alarm conditions consisting of logical operators. The alarm expression supports the following logical operators, which can be enclosed by "(" and ")". ● **+**: addition ● **-**: subtraction ● ***:** multiplying ● **/**: division ● **==**: equal to ● **!=**: not equal to ● **>**: greater than ● **<**: less than ● **>=**: greater than or equal to ● **<=**: less than or equal to ● **!**: non ● **||**: or ● **&&**: and ● **abs**: absolute value For example, if **Rule Template** of the source and destination of the comparison job is set to **Table Rows**, you can configure the alarm expression as follows: ● To configure an alarm to be generated when the number of rows in the source table is less than 100, enter **${1_1}<100**, where **${1_1}** indicates the total number of rows in the source table. ● To configure an alarm to be generated when the number of rows in the source table is not equal to that in the destination table, enter **${1_1}!= ${2_1}**, where **${1_1}** indicates the total number of rows in the source table and **${2_1}** indicates the total number of rows in the destination table. ● To configure an alarm to be generated when the number of rows in the source table is less than 100 or when the number of rows in the source table is not equal to that in the destination

Module	Parameter	Description
		table, enter **(${1_1}<100)\|\|(${1_1}!=${2_1})**, where **${1_1}** and **${2_1}** indicate the total number of rows in the source and destination tables, respectively, and **\|\|** indicates that an alarm is generated if either condition is met. • To configure an alarm to be generated when the absolute value of the number of rows in the source table minus the number of rows in the destination table divided by the number of rows in the source table is greater than 0.1, enter **abs(${1_1}-${2_1})/${1_1}>0.1**, where **${1_1}** and **${2_1}** indicate the total number of rows in the source and destination tables, respectively.
	Parameter	The value of this parameter is obtained from the output of the rule template. If you can click a parameter, the expression of the parameter is displayed in **Alarm Expression**. For example, if **Template** is set to **Table Rows**, **${1_1}** is displayed in **Alarm Expression** when you click alarm parameter **Table Rows**.

Module	Parameter	Description
	Logical Operator	This parameter is optional. You can perform logical operations on the result of an alarm expression to generate more complex alarm conditions. You can move the cursor between two alarm expressions in **Alarm Expression** and click one of the following operators to insert them. You can also manually enter an operator. The current expression supports the following logical operators which can be enclosed by brackets (). ● **+**: addition ● **-**: subtraction ● ***:** multiplying ● **/**: division ● **==**: equal to ● **!=**: not equal to ● **>**: greater than ● **<**: less than ● **>=**: greater than or equal to ● **<=**: less than or equal to ● **!**: non ● **||**: or ● **&&**: and ● **abs**: absolute value For example, if **Template** is **Table Rows** and if you want to generate an alarm when the number of rows in the source table is less than 100 or when the number of rows in the source table is not equal to that in the destination table, enter **(${1_1}<100)||(${1_1}!=${2_1})**, where **${1_1}** and **${2_1}** indicate the total number of rows in the source and destination tables, respectively, and **||** indicates that an alarm is generated if either condition is met.

5. Click **Next** and set the subscription configuration. If the SMN notification is required, enable **Notification**, and set **Notification Type** and **Topic**. See **Figure 7-40**.

Figure 7-40 Subscription configuration



NOTE

When notification is enabled, a notification is sent for all the subjobs of the configured notification type.

If you enable alarming, you do not need to set the notifications for failures. Alarms will be automatically reported if a task fails.

Currently, only SMS and email are available for subscribing to topics.

You can select **Alarm triggered** or **Run successfully** for **Notification Type**.

If you enable **Notification Policy**, you can set the condition for sending an alarm notification, that is, the number of consecutive alarms within a certain period of time (minutes). The period ranges from 1 minute to 360 minutes, and the number of consecutive times ranges from 1 to 10.

6. Click **Next** to go to the page where you can select a scheduling mode. Currently, **Once** and **On schedule** are supported. Set parameters for scheduling periodically by referring to **Table 7-16**. Click **Submit**.

NOTE

1. If **Once** is selected, a manual task instance is generated. A manual task has no dependency on scheduling and must be manually triggered.
2. If **On schedule** is selected, a periodic instance is generated. A periodic instance is an instance snapshot that is automatically scheduled when a periodic task reaches the scheduled execution time.
3. When a periodic task is scheduled once, an instance workflow is generated. You can perform routine O&M management on scheduled instance tasks, such as viewing the running status, stopping and rerunning the scheduled tasks.
4. Only MRS clusters that support job submission through an agency support periodic scheduling of comparison jobs. MRS clusters that support job submission through an agency are as follows:
   - Non-security MRS cluster
   - MRS security cluster whose version is later than 2.1.0, and that has MRS 2.1.0.1 or later installed

Table 7-16 Parameters for setting the scheduling mode

Parameter	Description
Effective	Effective date of a scheduling task.

Parameter	Description
Cycle	The frequency at which a scheduling task is executed. Related parameters are: • Minutes • Hours • Days • Weeks **NOTE** • If **Cycle** is set to **Minutes** or **Hours**, set the start time, end time, and interval for the scheduling task. • If **Cycle** is set to **Days**, set the start time of the scheduling task. • If **Cycle** is set to **Weeks**, set **Scheduling Time** and **Start from** for the scheduling task, that is, *XX* o'clock *XX* minutes on *XXX* every week.

After a comparison job is created, you can view it in the job list. You can also filter jobs by job name, creator, and time range. The system supports fuzzy search.

After a comparison job is created, you can edit, delete, run, start scheduling, and stop scheduling it.

NOTE

You cannot start scheduling a one-off comparison job.

Running a Comparison Job

To run a comparison job, perform the following operations:

Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **Comparison Jobs**. In the right pane, locate a comparison job.

Step 2 Click **Run** in the **Operation** column.

Step 3 In enterprise mode, select the development environment or production environment.

Step 4 Click **OK**.

----End

Exporting Comparison Jobs

You can export a maximum of 200 comparison jobs. Each cell of the exported file can contain a maximum of 65,534 characters.

Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **Comparison Jobs**. In the right pane, select the comparison jobs to export.

Step 2 Click **Export**. The **Export Comparison Job** dialog box is displayed.

Step 3 Click **Export** to switch to the **Export Records** tab.

Step 4 In the list of exported files, locate an exported comparison job and click **Download** in the **Operation** column to download the Excel file of the job to the local PC.

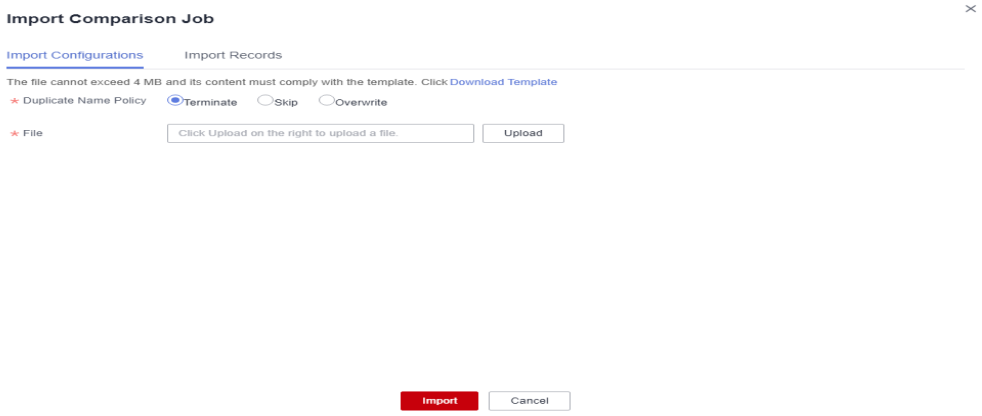
----End

Importing Comparison Jobs

You can import a file containing a maximum of 4 MB data. Each cell of the file to be imported can contain a maximum of 65,534 characters.

Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **Comparison Jobs**. In the right pane, click **Import**. The **Import Comparison Job** dialog box is displayed.

Figure 7-41 Importing comparison jobs



Step 2 On the **Import Configurations** tab page, set **Duplicate Name Policy**.

- **Terminate**: If comparison job names repeat, all comparison jobs will fail to be imported.
- **Skip**: If comparison job names repeat, the comparison jobs will still be imported.
- **Overwrite**: If comparison job names repeat, new jobs will replace existing ones with the same names.

  NOTE

  If you select **Overwrite**, stop job scheduling before uploading a file. Otherwise, the upload will fail.

Step 3 Click **Upload** and select the prepared data file.

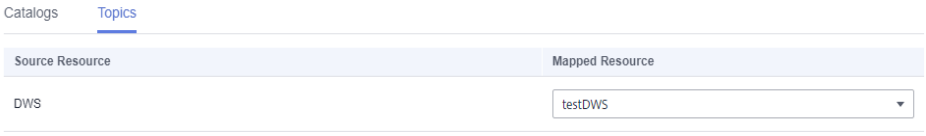
NOTE

Edit the data file using either of the following methods:

- (Recommended) Click **Export** to export data and import the data to the system directly or import it after modification.
- Click **Download Template**, fill in the template with the data to import, and import the data to the system.

Step 4 Configure resource mapping for the data connection, cluster, directory, and topic. If you do not configure the resource mapping, the original mapping is used by default.

Figure 7-42 Configuring the resource mapping



- **Data Connection**: Select the type of the imported data connection.
- **Cluster**: If the data connection type is DLI, select the corresponding queue.
- **Directory**: Select the directory where the imported comparison job is stored.
- **Topic**: If SMN is configured, you need to select a topic.

Step 5 Click **Import** to import the Excel template to the system.

Step 6 Click the **Import Records** tab to view the import records.

----End

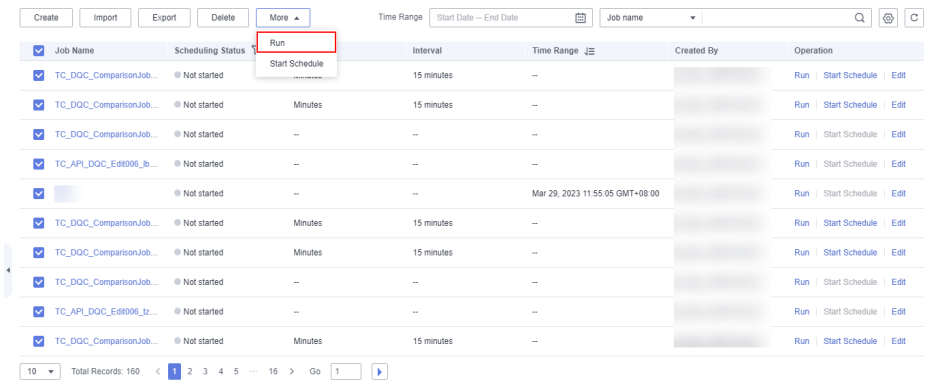
Running Comparison Jobs

You can run a maximum of 200 comparison jobs at a time.

Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **Comparison Jobs**. In the right pane, select the comparison jobs you want to run.

Step 2 Above the job list, click **More** and select **Run** to run the selected comparison jobs.

Figure 7-43 Running jobs



Step 3 In enterprise mode, select the development environment or production environment.

Step 4 Click **OK**.

----End

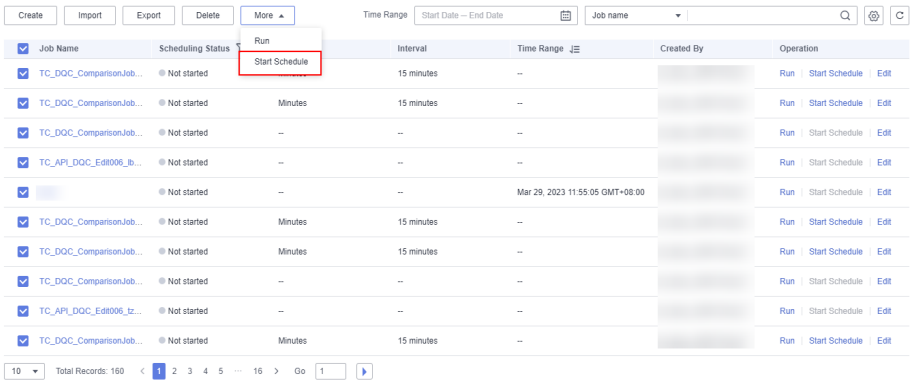
Scheduling Comparison Jobs

You can schedule a maximum of 200 comparison jobs at a time.

Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **Comparison Jobs**. In the right pane, select the comparison jobs you want to schedule.

Step 2 Above the job list, click **More** and select **Start Schedule** to schedule the selected comparison jobs.

Figure 7-44 Scheduling jobs



----End

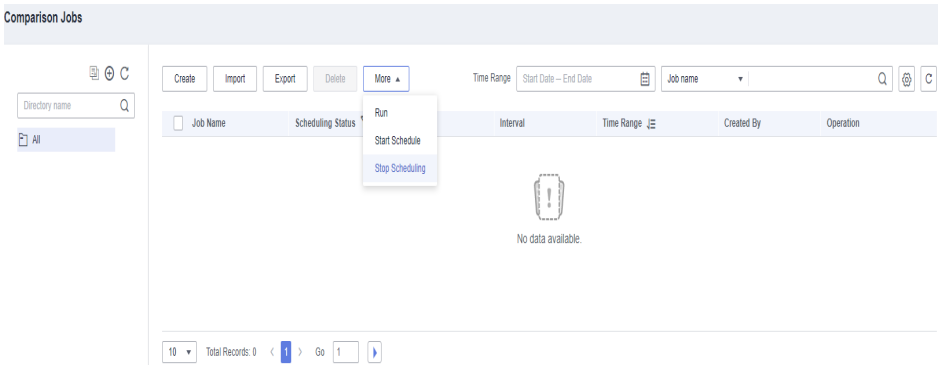
Stopping Scheduling Comparison Jobs

You can stop scheduling a maximum of 200 comparison jobs at a time.

Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **Comparison Jobs**. In the right pane, select the comparison jobs you want to stop scheduling.

Step 2 Above the job list, click **More** and select **Stop Scheduling** to stop scheduling the selected comparison jobs.

Figure 7-45 Stopping scheduling jobs



----End

Stopping Comparison Jobs

You can stop a maximum of 200 comparison jobs at a time.

Only comparison jobs in **Running** state can be stopped.

Step 1 In the left navigation pane on the **DataArts Quality** page, choose **Quality Monitoring** > **O&M**. In the right pane, select the comparison jobs you want to stop.

Step 2 Click **Stop**. In the displayed **Stop Instance** dialog box, confirm the instances to stop and click **Yes**.

Figure 7-46 Stopping instances

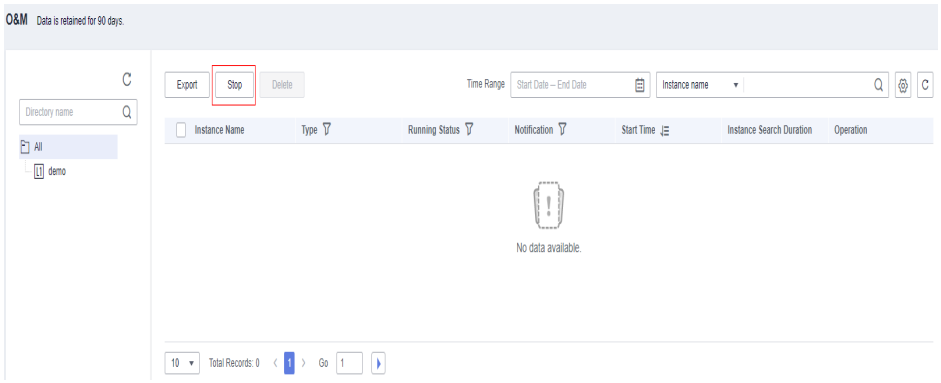
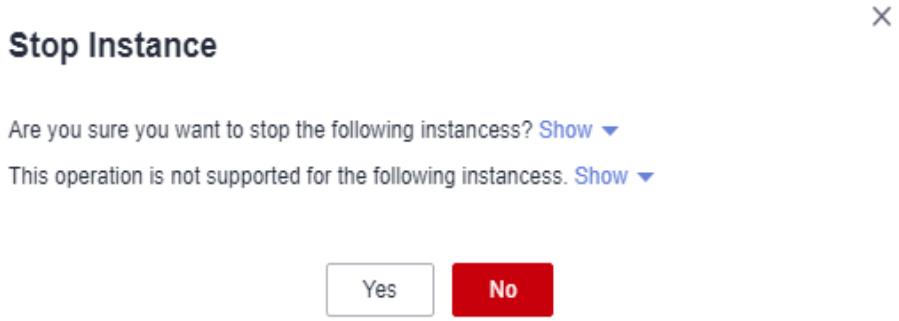


Figure 7-47 Stopping instances



----End

7.2.5 Viewing Job Instances

GUI Description

The following figure shows the areas and buttons on the **O&M** page.

Figure 7-48 O&M page

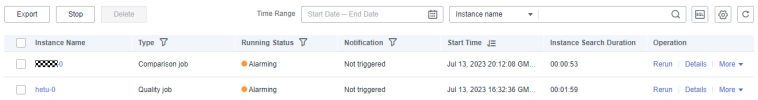


Table 7-17 O&M page

No.	Area	Description
1	Navigation bar	Contains the storage directory of data quality rules. You can store rules in different directories tailored to service requirements. The number next to each directory indicates the number of rule instances stored in the directory.
2	List of rule instances	Displays the instance name, type, running status, and running result.
3	Management area	Provides buttons for exporting, deleting, and stopping selected instances.
4	Search area	• Displays rule instances based on specified conditions. For example, you can display rule instances for a specified time range. • Displays a list of instances according to the handler or instance name. Fuzzy search is supported.

No.	Area	Description
5	Concurrent SQL statements	Click **SQL**. In the displayed **Configure Concurrent SQL Statements for a Connection** dialog box, set the number of concurrent SQL statements. Enter a value from 10 to 1,000. Click **OK**. **NOTE** The concurrency refers to the number of concurrent SQL statements for a data connection.

Table 7-18 List of rule instances

Parameter	Description
Instance Name	Consists of a rule name and a number. The larger the number is, the later the instance is created.
Type	Displays the job type. The value can be **Quality Job** or **Comparison Job**.
Running Status	Displays the running status of an instance, such as **Successfully**, **Failed**, **Running**, and **Alarming**. In the right pane, you can view the detailed run logs of the rule instances. • **Successfully**: The instance stops normally and the running result meets the expectation. • **Failed**: The instance stops unexpectedly. • **Alarming**: The instance stops normally, but the running result does not meet the expectation. • **Running**: The instance is running, but no running result is displayed.
Notification	Displays the notification status of an instance, such as **Successfully**, **Failed**, and **Not triggered**.
Operator	Displays the operator of the instance.
Created	Displays the time when the instance was created.
Start Time	Displays the time when the instance starts to run. The start time can be sorted in ascending or descending order.
Running Duration	Displays the running duration of the instance.
End Time	Displays the time when the instance execution is complete. The end time can be sorted in ascending or descending order.
Handled By	Displays the handler of the instance.
Rerun	Allows you to run a rule instance again.

Parameter	Description
Details	Displays the running results and logs of job instances. • **Comparison Job Result** In the running result, the left pane displays the execution result of the rule for source table rows, and the right pane displays the execution result of the rule for destination table rows. The error rate indicates the difference between the number of rows of the source and destination tables. If the error rate is 0, the source and destination tables have the same number of rows.
More > Rectify	Allows you to perform further processing on a rule instance. For example, you can **Provide defects directly**, **Close defects**, or **Specify a user to rectify fault**. The above operations can be performed only when you are the handler of the instance.
More > Refresh Job Status	Refresh the job status.

7.2.6 Viewing Quality Reports

You can query the quality reports of business metrics and data objects to determine whether their quality meets the requirements.

NOTE

Quality reports include technical reports and business reports.

Technical reports measure the execution results of quality jobs and contain data connections, databases, table names, and scores.

Business reports measure the execution results of quality jobs associated with subjects in DataArts Architecture and contain subject area groups, subject areas, business objects, table names, and scores.

Viewing Data Quality Scores in a Technical Report

The full quality score can be set to 5, 10, or 100 points. By default, a five-point scale is used for quality scoring based on table-associated rules. Scores in different dimensions, such as tables and databases, are calculated based on the weighted average values of rule scores in different dimensions.

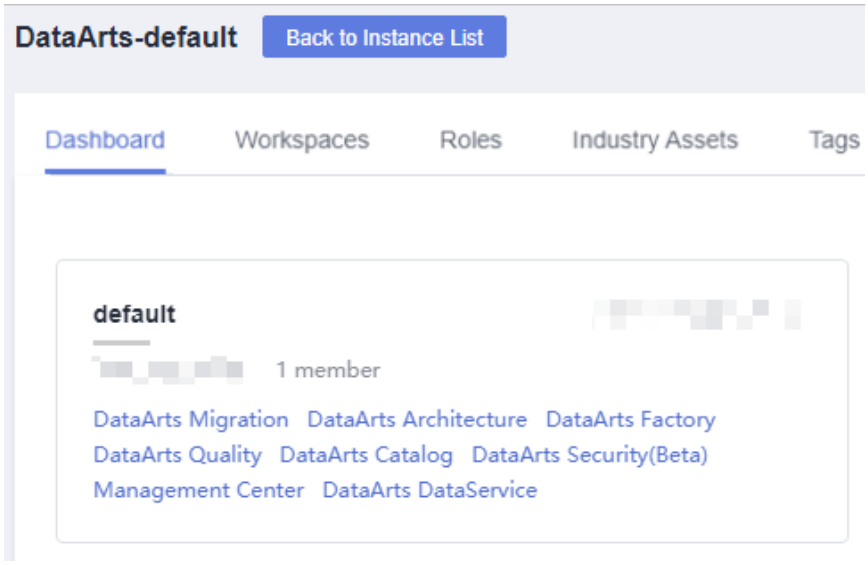
You can query the scores of databases, tables, and table-associated rules. For details on the calculation formulas, see **Table 7-19**.

Table 7-19 Formulas for calculating scores

Object	Formula
Rule	When a quality job that contains a percentage-related rule (either built-in or custom) is created, a quality report can be generated. • Percentage-related rules can be classified into positive rules and negative rules. For a positive rule, the higher the percentage is, the better the data quality is. For a negative rule, the higher the percentage is, the poorer the data quality is. Rules that contain the unique value percentage, duplicate value percentage, and valid percentage are positive rules, and rules that contain the null value percentage are negative rules. • Positive rule score = Number of data rows that meet the rule/Total number of data rows x 5. • Negative rule score = (1 - Number of data rows that meet the rule/Total number of data rows) x 5.
Table	The table score is calculated as follows: ∑(Scores of all rules associated with the table x Rule weight)/∑Rule weight.
Database	Weighted average value of the scores of all data tables in the database, that is, ∑Scores of all data tables in the database/Number of tables.
Data connection	Weighted average value of the scores of all databases in the data connection, that is, ∑Scores of all databases in the data connection/Number of databases.

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Quality**.

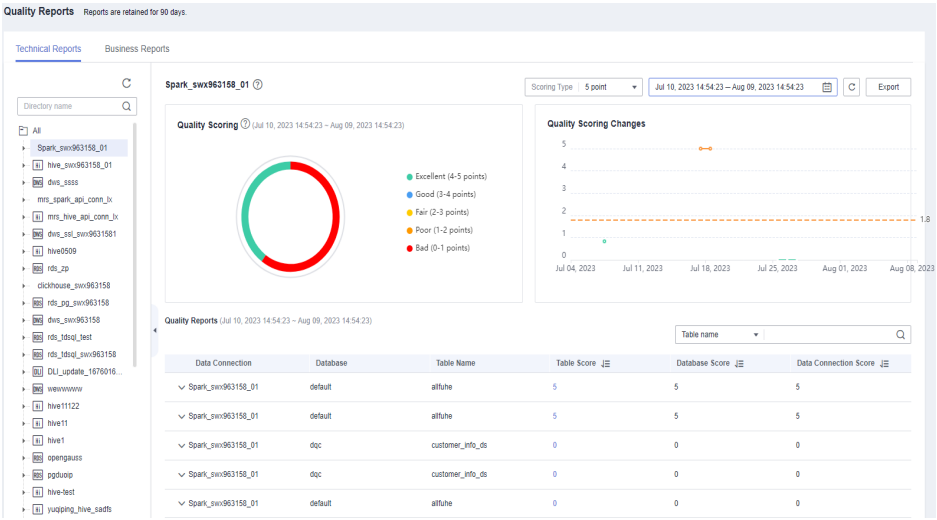
Figure 7-49 DataArts Quality



Step 2 Choose **Quality Monitoring** > **Quality Job** in the left navigation bar.

Step 3 On the **Technical Reports** page, select a data connection and set a time range (a maximum of 30 days).

Figure 7-50 Selecting a data connection

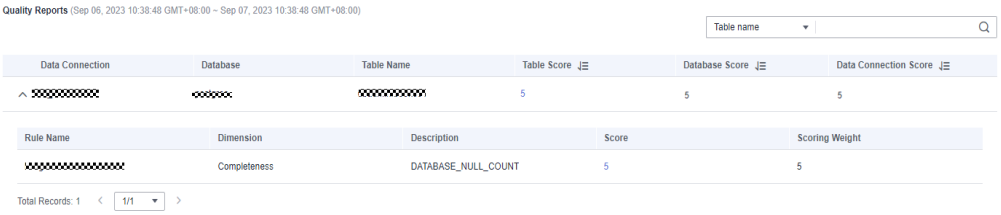


NOTE

- Take the full score 5 points as an example. Points 4 to 5: excellent; 3 to 4: good; 2 to 3: unqualified; 1 to 2: poor; 0 to 1: very poor.
- The quality score data of a day is generated in the early morning of the next day.
- In the **Quality Scoring Changes** area, the solid line consists of the quality scores of the end date and the previous seven days, and the dashed line indicates the average quality score of these days.
- If the job is executed multiple times on a day, the last score is used as the quality score of the day.

Step 4 Click the score link in the **Table Score** column to expand the scores of the rules associated with the table.

Figure 7-51 Viewing the rule score

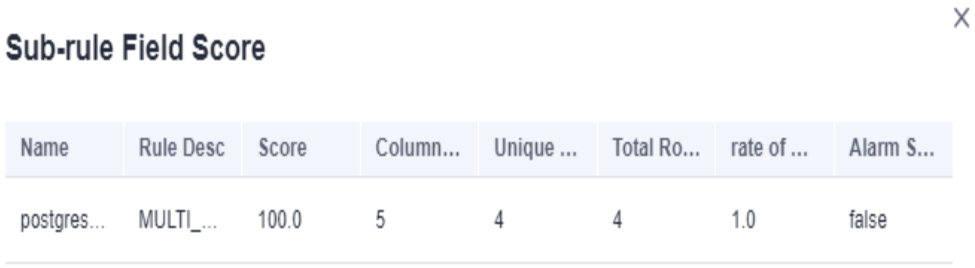


NOTE

The rule name is the name of the running instance. If a job runs multiple times, the name of the latest instance is used. If a running instance contains multiple sub-instances, each sub-instance has a record.

Step 5 Click the score link in the **Rule Score** column to expand the scores of the fields associated with the rule.

Figure 7-52 Table-associated rule scores



----End

Viewing Business Quality Scores in a Business Report

The full quality score can be set to 5, 10, or 100 points. By default, a five-point scale is used for quality scoring based on table-associated rules. The scores in different dimensions, such as tables, business objects, and subject areas, are calculated based on the weighted average values of rule scores in different dimensions.

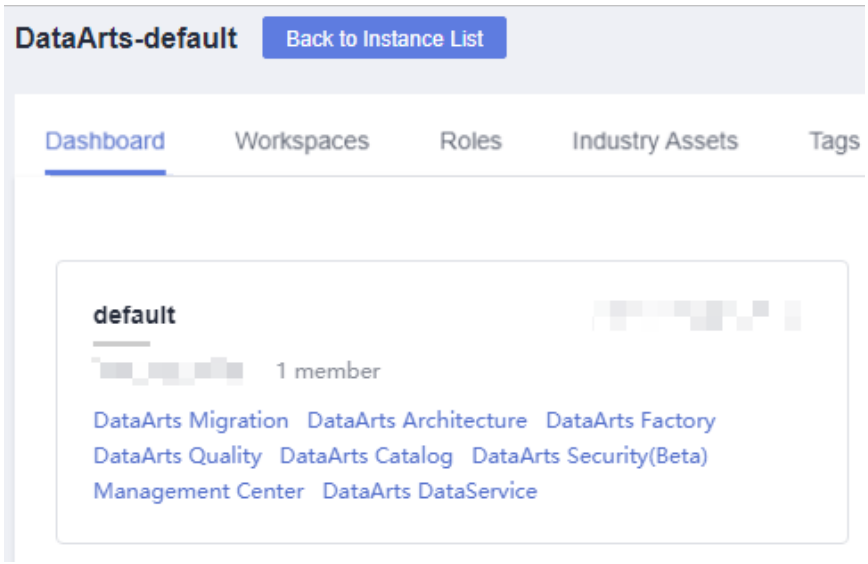
You can query the quality scores of subject area groups, subject areas, business objects, tables, and table-associated rules. For details on the calculation formulas, see **Table 7-20**.

Table 7-20 Formulas for calculating scores

Object	Formula
Rule	When a quality job that contains a percentage-related rule (either built-in or custom) is created, a quality report can be generated. ● Percentage-related rules can be classified into positive rules and negative rules. For a positive rule, the higher the percentage is, the better the data quality is. For a negative rule, the higher the percentage is, the poorer the data quality is. Rules that contain the unique value percentage, duplicate value percentage, and valid percentage are positive rules, and rules that contain the null value percentage are negative rules. ● Positive rule score = Number of data rows that meet the rule/ Total number of data rows x Full score (5, 10, or 100 points). ● Negative rule score = (1 – Number of data rows that meet the rule/Total number of data rows) x Full score (5, 10, or 100 points). ● If the table is empty (the total number of rows is 0), the positive rule score is fixed at the full score and the negative rule score is fixed at 0 points.
Table	The table score is calculated as follows: ∑(Scores of all rules associated with the table x Rule weight)/∑Rule weight.
Business object	Weighted average value of the scores of all tables under the business object, that is, ∑Scores of all tables under the business object/Number of tables.
Subject area	Weighted average value of scores of all business objects in the subject area, that is, ∑Scores of all business objects in the subject area/Number of business objects.
Subject area group	Average weighted value of the scores of all subject areas in the group, that is, ∑Scores of all subject areas in the group/Number of subject areas.

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Quality**.

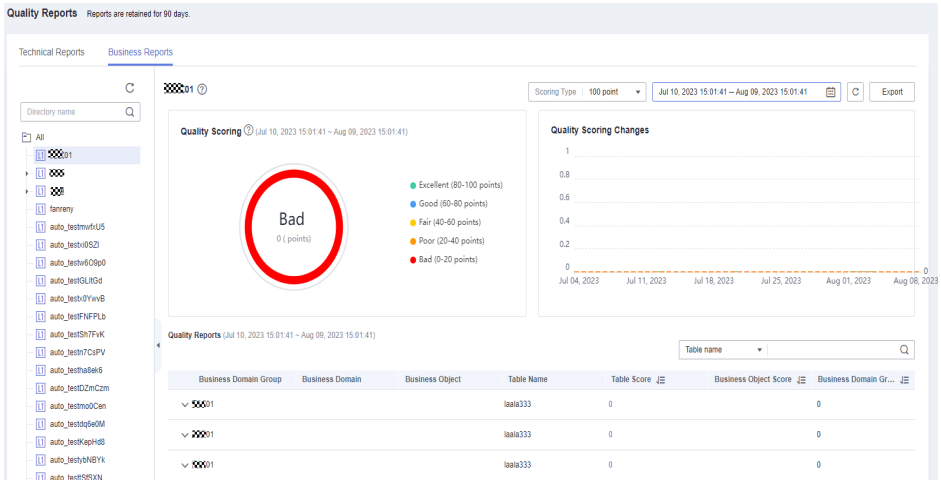
Figure 7-53 DataArts Quality



Step 2 Choose **Quality Monitoring** > **Quality Job** in the left navigation bar.

Step 3 Click the **Business Reports** tab, and select a subject and an end date to query the quality scores of the end date and the previous seven days, as shown in **Figure 7-54**.

Figure 7-54 Business object



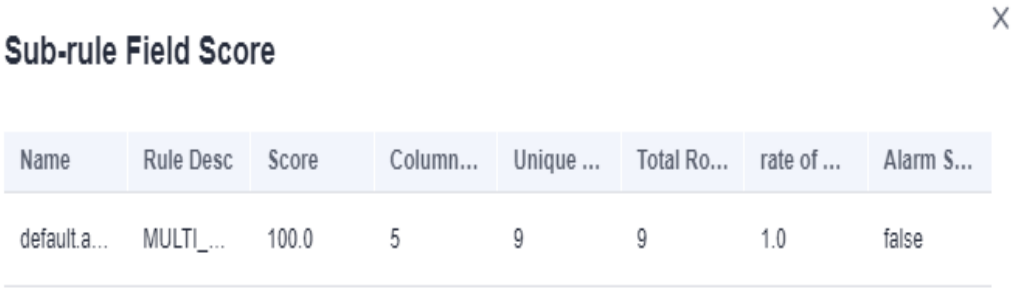
NOTE

- Take the full score 5 points as an example. Points 4 to 5: excellent; 3 to 4: good; 2 to 3: fair; 1 to 2: qualified; 0 to 1: unqualified.
- The quality score data of a day is generated in the early morning of the next day.
- In the **Quality Scoring Changes** area, the solid line consists of the quality scores of the end date and the previous seven days, and the dashed line indicates the average quality score of these days.
- If the job is executed multiple times on a day, the last score is used as the quality score of the day.

Step 4 Click the score link in the **Table Score** column to expand the scores of the rules associated with the table.

Step 5 Click the score link in the **Rule Score** column to expand the scores of the fields associated with the rule.

Figure 7-55 Table-associated rule scores



----End

Exporting Quality Reports

You can export a quality report in either of the following ways:

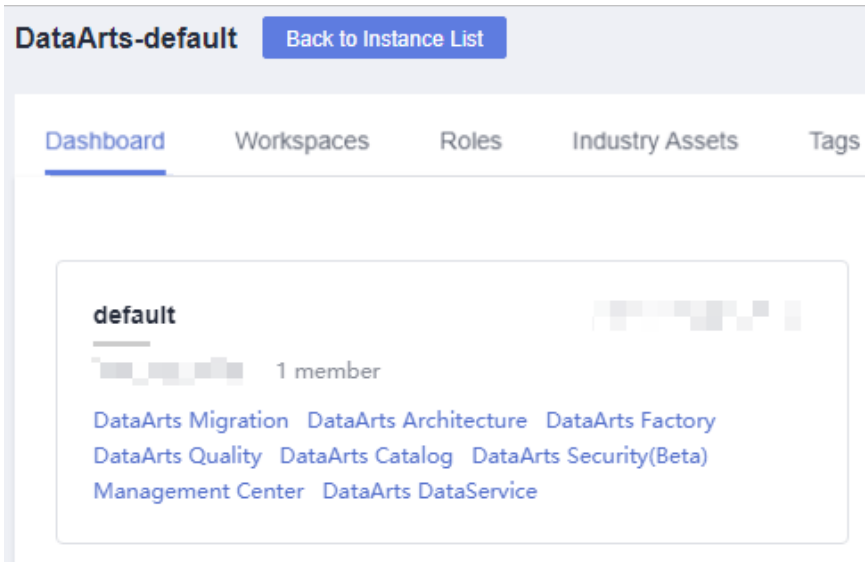
- If the OBS service is available, the data is exported to the associated OBS bucket by default.

  NOTE

  - As quality reports contain a large amount of data, a single exported file can contain a maximum of 2,000 fields. Therefore, there may be multiple exported files in the OBS bucket.
  - The exported report is available only in the current workspace.
- If the OBS service is unavailable, the data is exported to a local path by default.

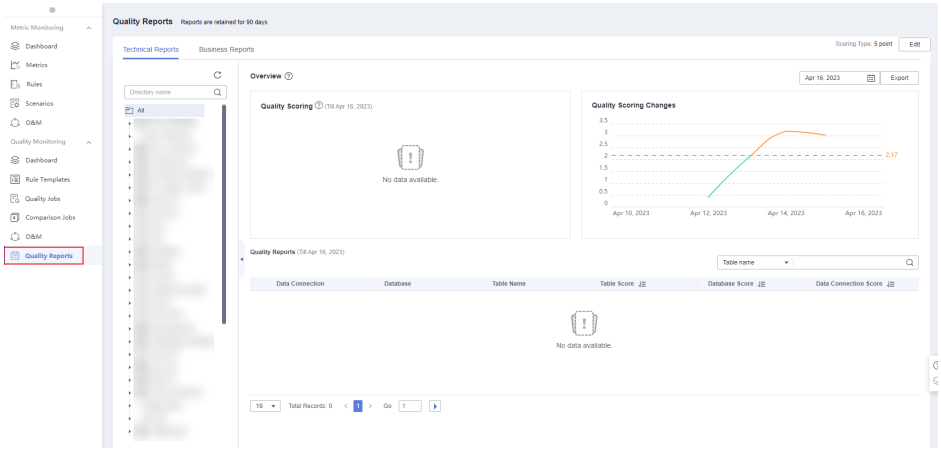
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Quality**.

Figure 7-56 DataArts Quality



Step 2 Choose **Quality Monitoring** > **Quality Job** in the left navigation bar.

Figure 7-57 Quality Reports page



Step 3 In the upper right corner of the page, click **Export**.

Figure 7-58 Export

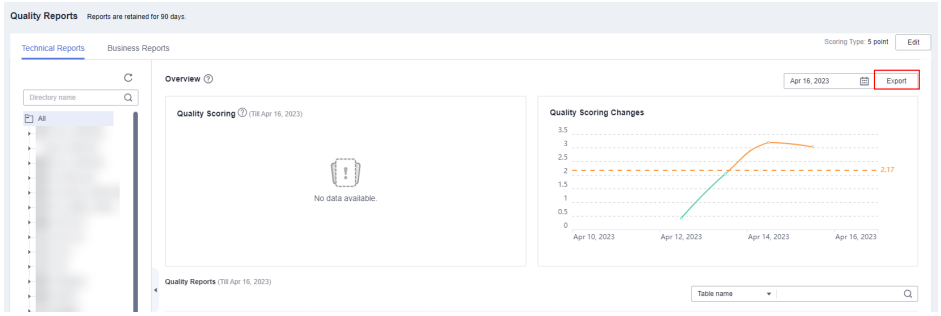
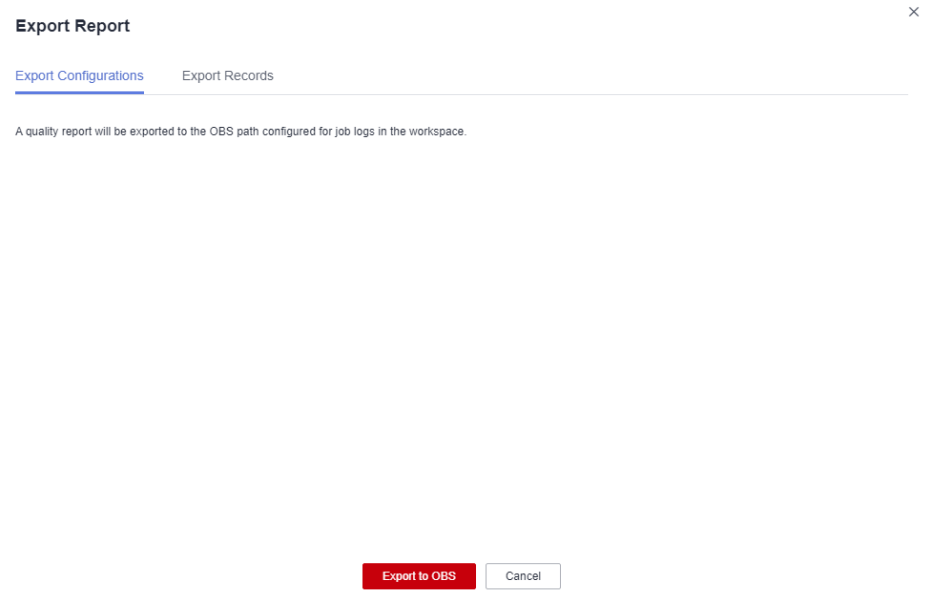
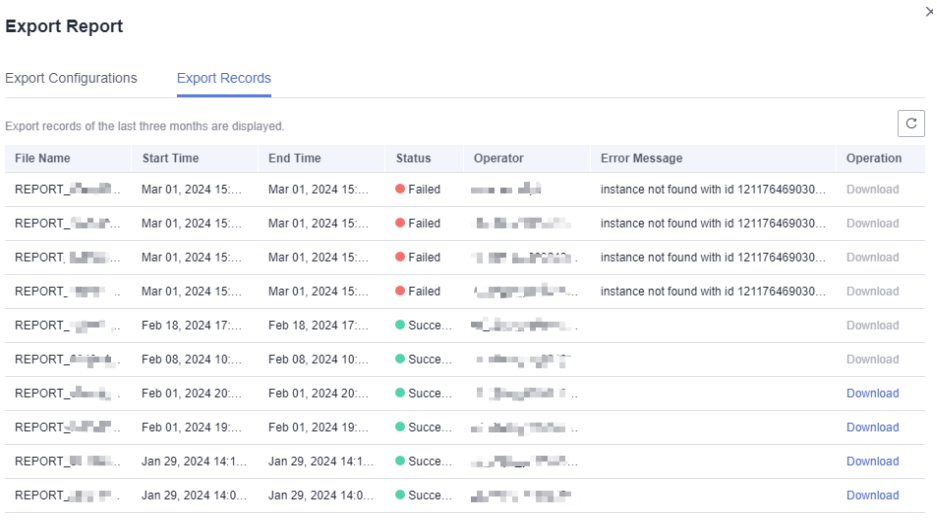


Figure 7-59 Export to OBS



Step 4 Click the **Export Records** tab to view the export result. You can click **Download** to download a report. If the exported report file is too large, you can directly download the file.

Figure 7-60 Export Records



----End

7.3 Tutorials

7.3.1 Creating a Business Scenario

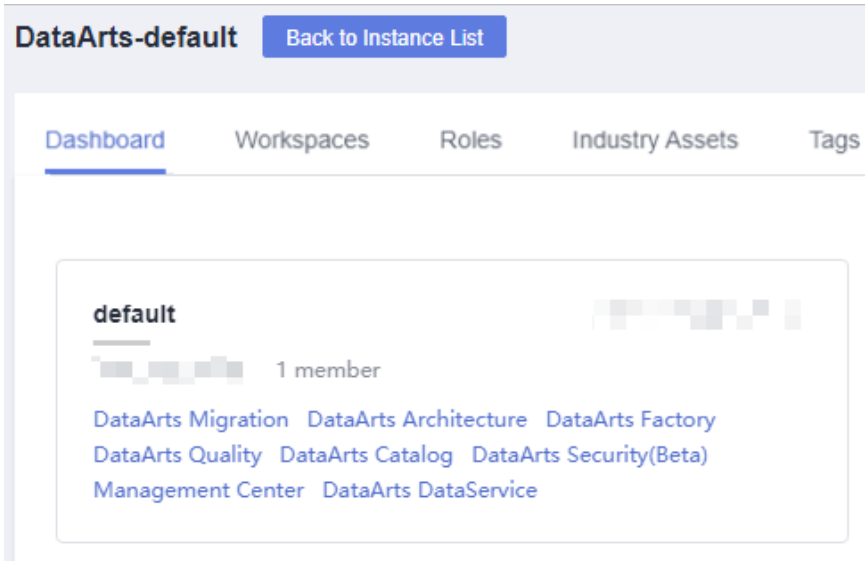
Scenario

Business scenarios are used to monitor business metrics. This section describes how to create a business scenario.

Procedure

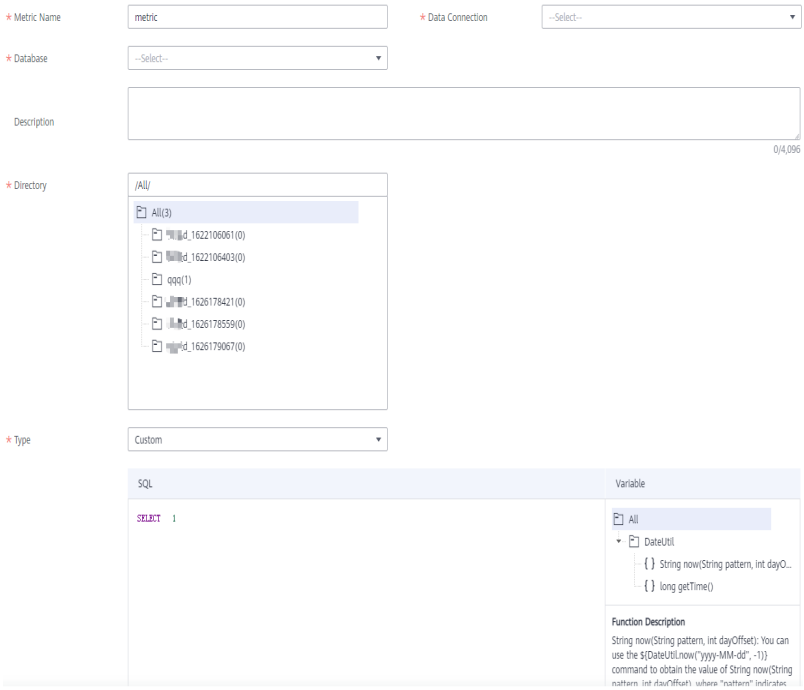
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Quality**.

Figure 7-61 DataArts Quality



Step 2 Create a metric.

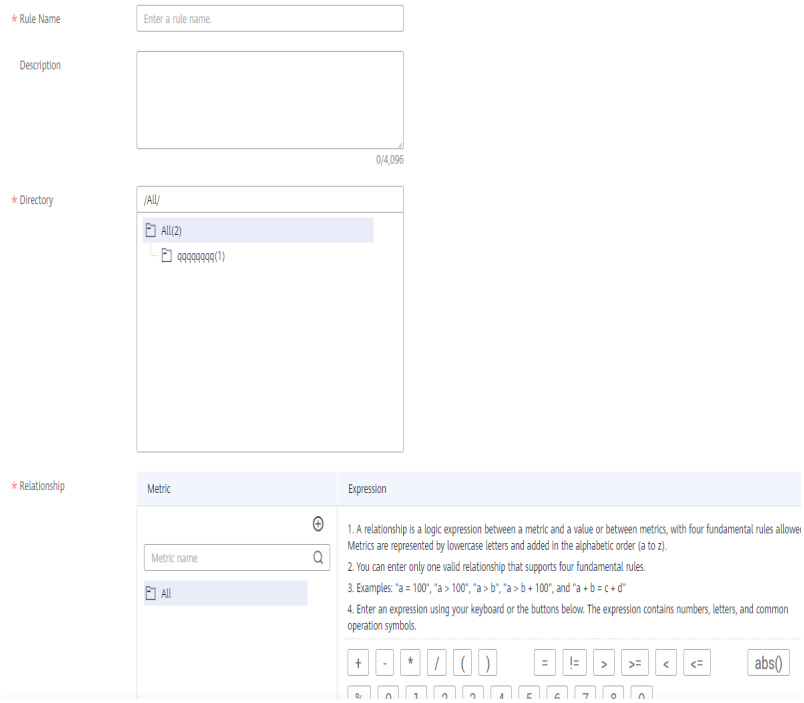
1. In the navigation pane on the left, choose **Metrics**.
2. On the **Metrics** page, click **Create**.



3. Click **Trial Run** to check whether the metric runs properly.
4. Click **OK**.

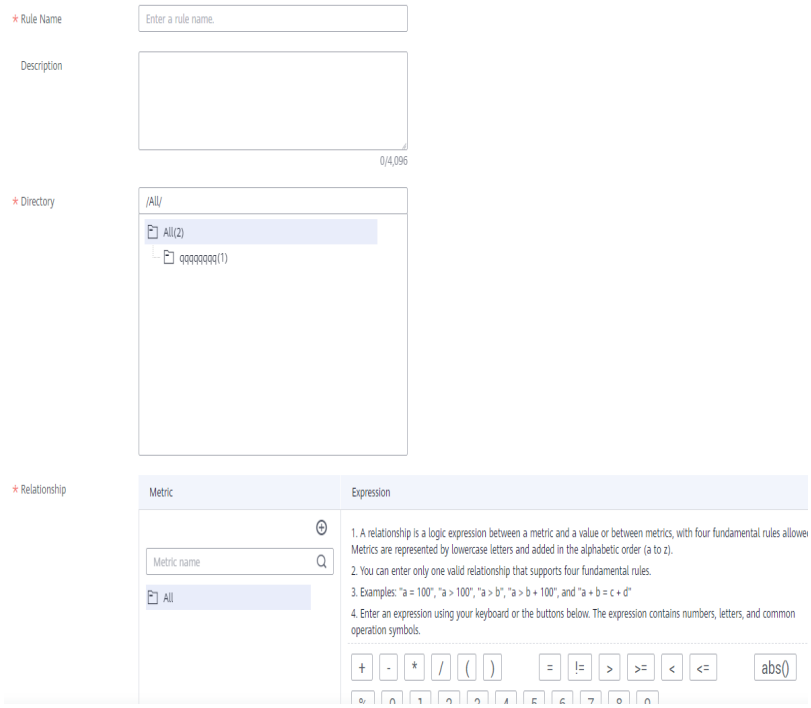
Step 3 Create a rule.

1. In the navigation pane on the left, choose **Rules**.
2. On the **Rules** page, click **Create**.
3. Set the parameters shown in the following figure.



4. Click **OK**.
5. On the **Rules** page, click **Create** to create another rule.

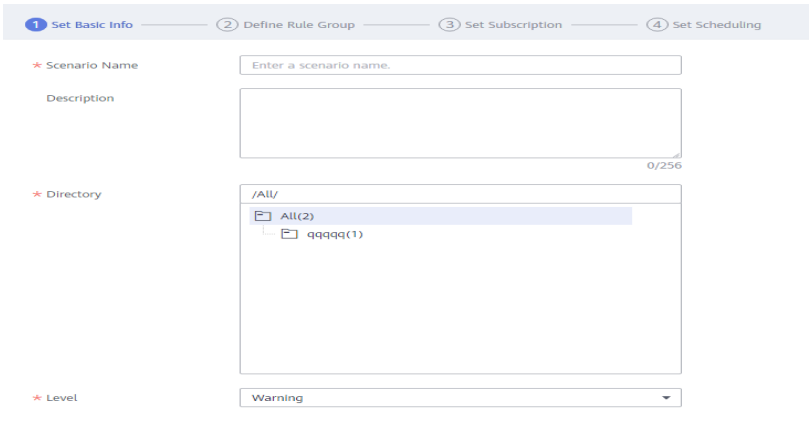
6. Set the parameters shown in the following figure.



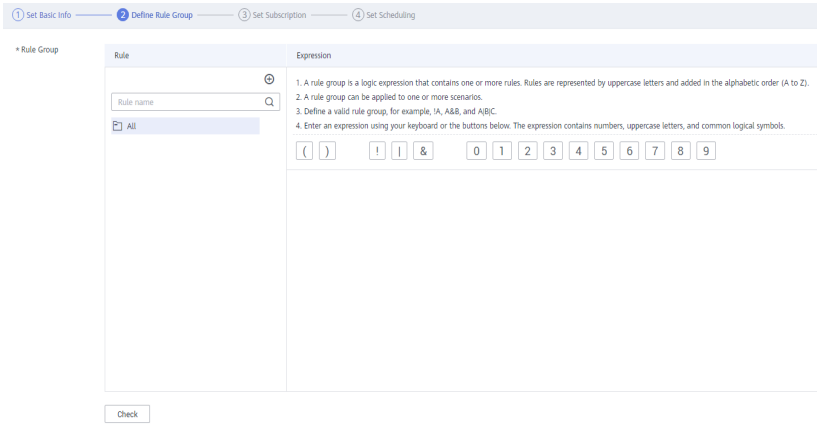
7. Click **OK**.

Step 4 Create a scenario.

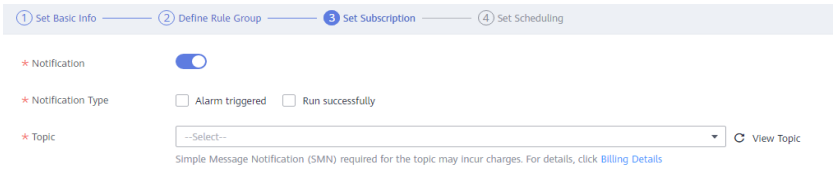
1. In the navigation pane on the left, choose **Scenarios**.
2. On the **Scenarios** page, click **Create**. On the displayed **Create Scenario** page shown in the following figure, set the required parameters.



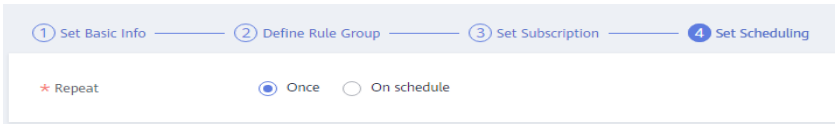
3. Click **Next** and set the parameters for the rule group.



4. Click **Next** and set subscription parameters.



5. Click **Next** and set scheduling parameters.



6. Click **Submit**.

Step 5 In the scenario list, locate the created scenario and click **Run** in the **Operation** column.

1. Click the refresh button in the upper right corner. The **Running Status** of the scenario is **Succeeded**.
2. Click the running result to view details.

----End

7.3.2 Creating a Quality Job

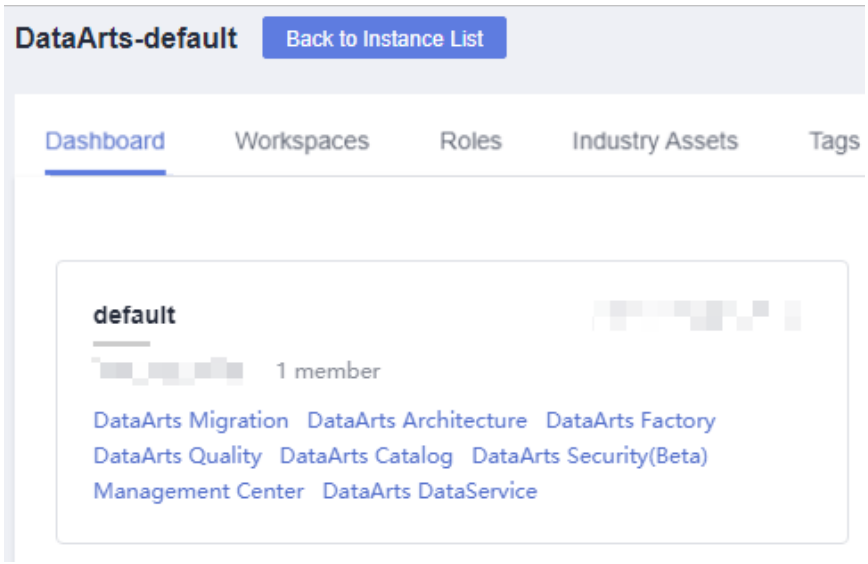
Scenario

You can use a quality job to monitor data quality. This section describes how to create a quality job.

Procedure

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Quality**.

Figure 7-62 DataArts Quality



Step 2 Create a rule template.

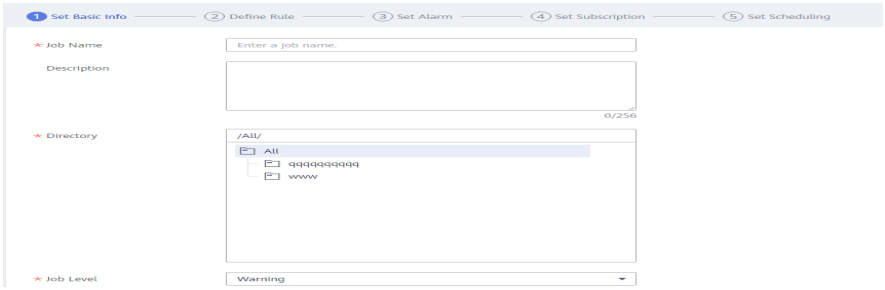
1. In the navigation pane on the left, choose **Rule Templates**. System templates are displayed. Rule templates have six dimensions: completeness, uniqueness, timeliness, validity, accuracy, and consistency.
2. **Optional:** Click **Create** to create a rule template.

   NOTE

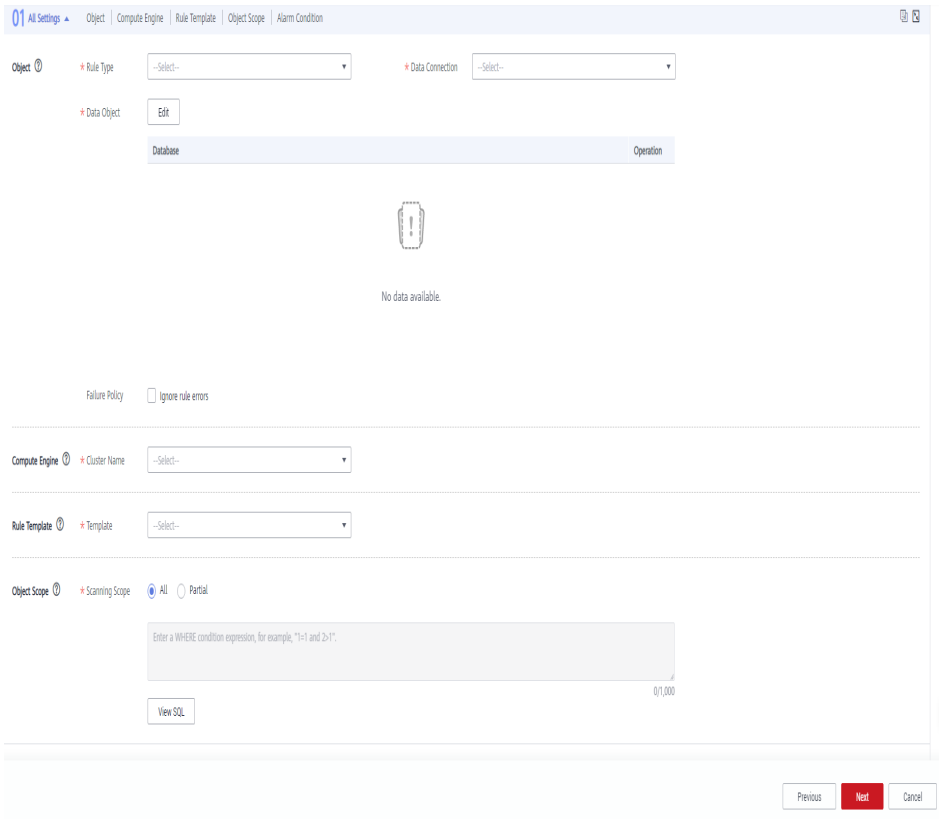
   In this example, use a system rule.

Step 3 Create a quality job.

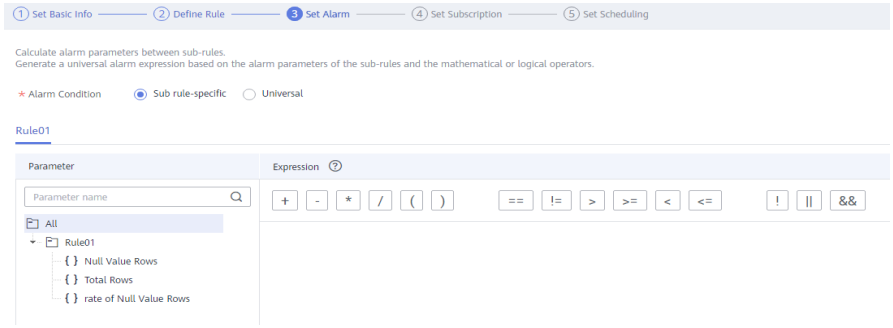
1. In the navigation pane on the left, choose **Quality Jobs**.
2. Click **Create**. On the **Create Quality Job** page, set basic information about the quality job.



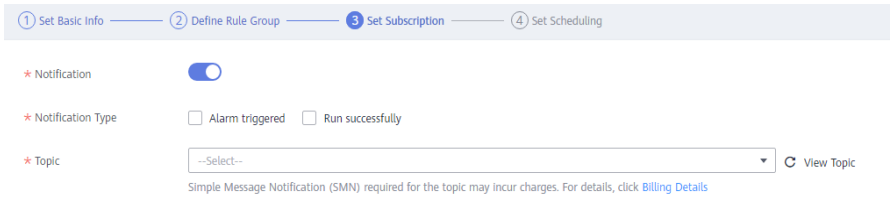
3. Click **Next** to go to the **Define Rule** page. Click on the rule card to configure the rule.



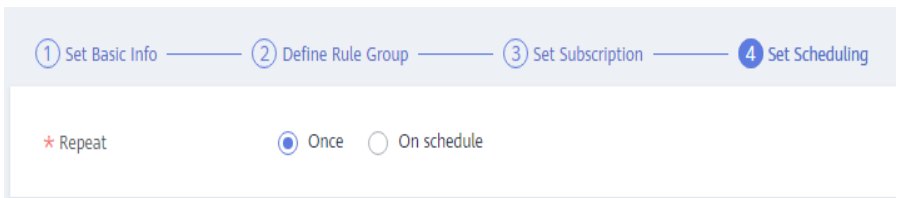
4. Click **Next** and set alarm parameters.



5. Click **Next** and set subscription parameters.



6. Click **Next** and set scheduling parameters.

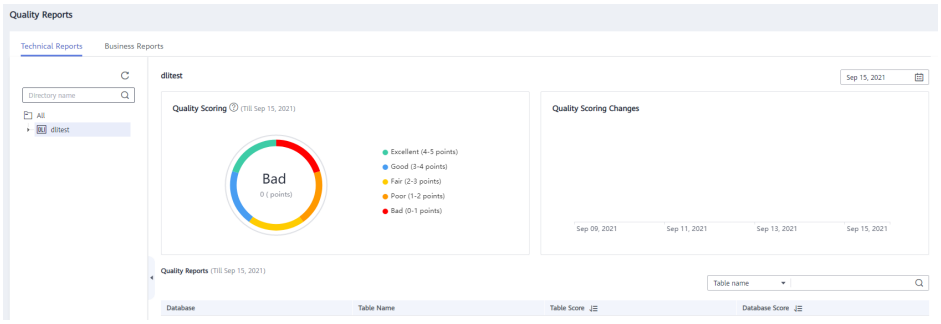


7. Click **Submit**.

Step 4 In the quality job list, locate the created job and click **Run** in the **Operation** column.

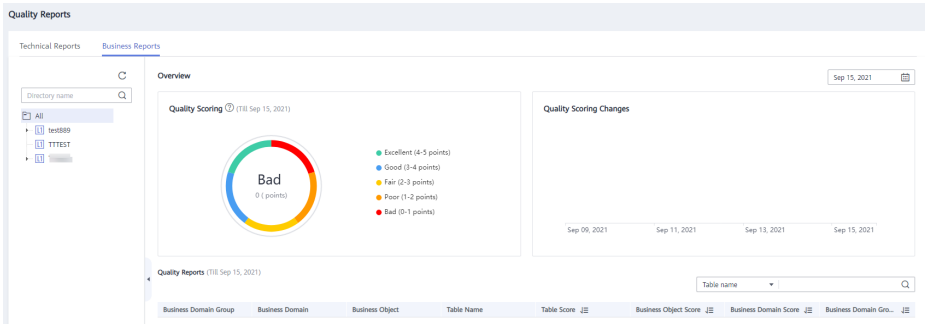
1. After the quality job is successfully run, choose **Quality Reports** in the navigation pane on the left.
2. The **Technical Reports** page is displayed by default.

Figure 7-63 Technical report



3. Click the **Business Reports** tab and view the business reports.

Figure 7-64 Business report



----End

7.3.3 Creating a Comparison Job

Scenario

Data comparison is critical to ensure data consistency in data development and migration. The cross-source data comparison capability is the key to checking consistency of the data before and after migration or processing. This section describes how to create a comparison job in the DataArts Quality module of DataArts Studio to verify consistency between a DLI and DWS connection.

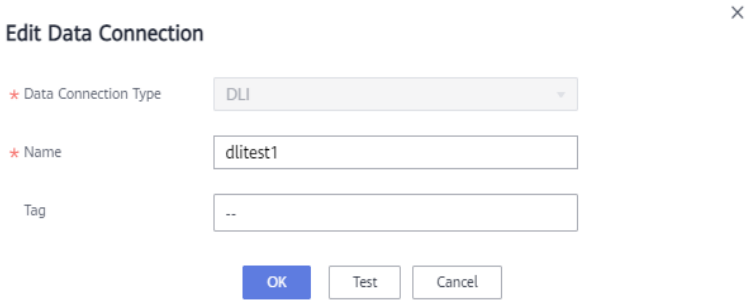
Environment Preparations

Create the data sources to compare, that is, create different types of data connections in the Management Center.

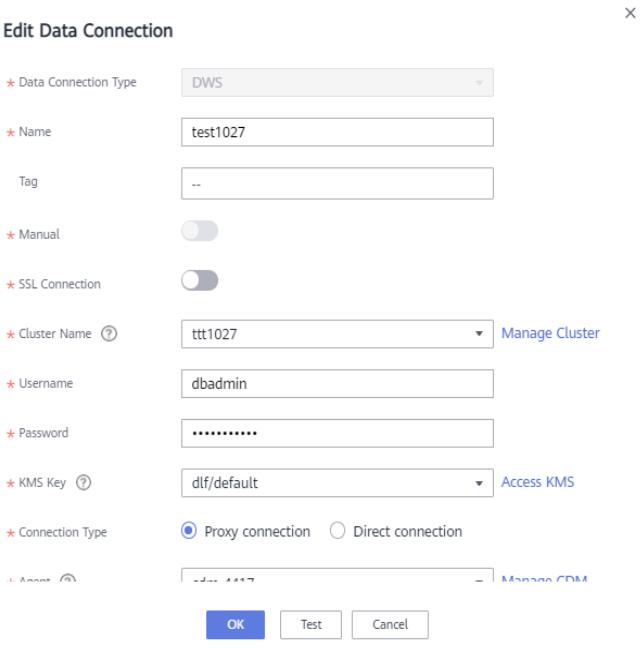
Procedure

Step 1 Create different types of data connections.

1. Create a DLI data connection. On the **Management Center** page, click **Create Data Connection**. In the displayed dialog box, select **DLI** for **Data Connection Type**, enter a connection name, and click **Test**. If the message "Connected." is displayed, click **OK**.



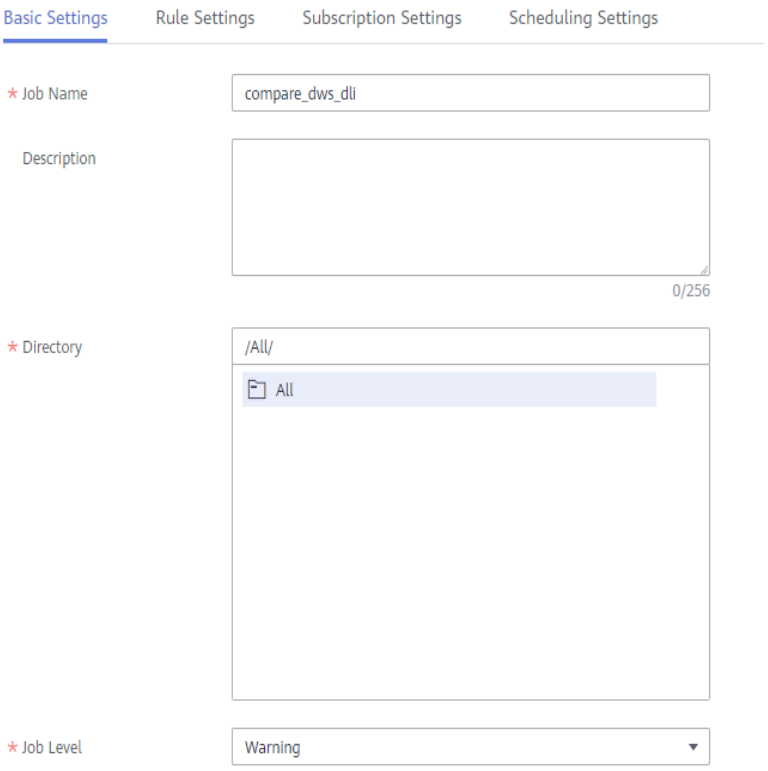
2. Create a DWS data connection. On the **Management Center** page, click **Create Data Connection**. In the displayed dialog box, select **DWS** for **Data Connection Type**, enter a connection name, set other required parameters, and click **Test**. If the message "Connected." is displayed, click **OK**.



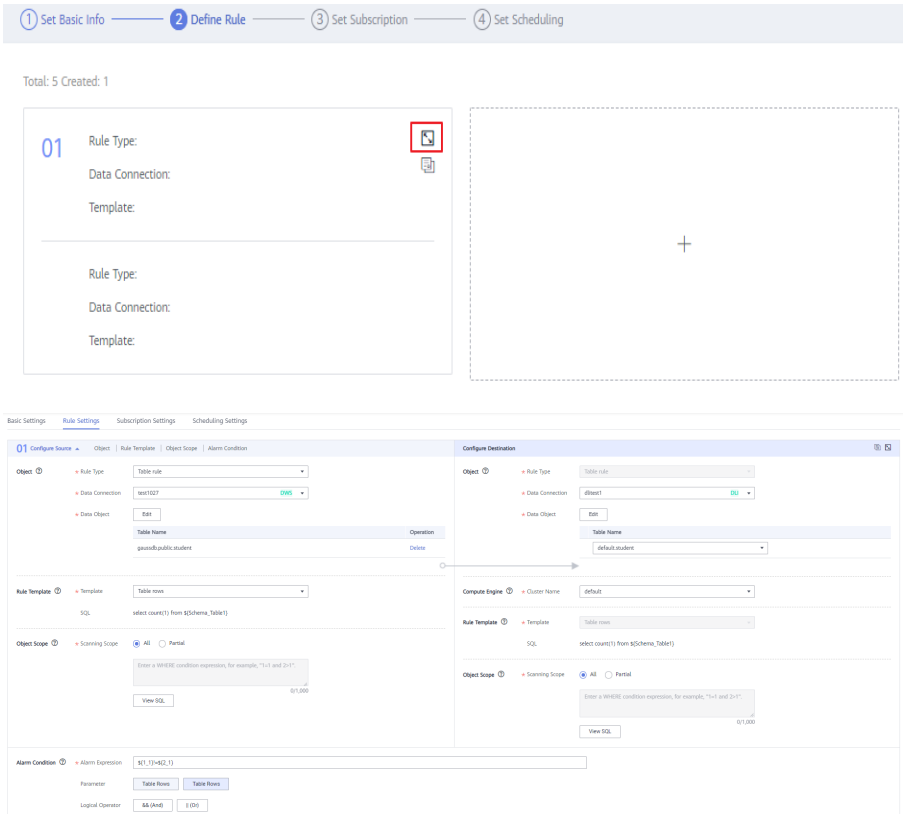
Step 2 Create a comparison job.

1. On the **DataArts Quality** page, choose **Comparison Jobs** in the navigation pane.
2. Click **Create**. On the **Create Comparison Job** page, set basic information about the comparison job.

Figure 7-65 Configuring basic information



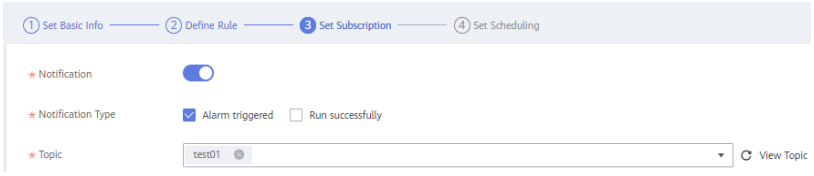
3. Click **Next** to go to the **Define Rule** page. Click on the rule card to configure the rule.



NOTE

- You need to configure information about both the source and destination. For how to configure the source connection, see **Configuring a DWS Connection**. For how to configure the destination connection, see **Configuring a DLI Connection**.
- When configuring **Alarm Condition**, **${1_1}** indicates the number of rows in the source table, and **${2_1}** indicates the number of rows in the destination table. In the preceding figure, the alarm condition **${1_1}!=${2_1}** indicates that an alarm is generated when the number of rows in the source table is inconsistent with that in the destination table.

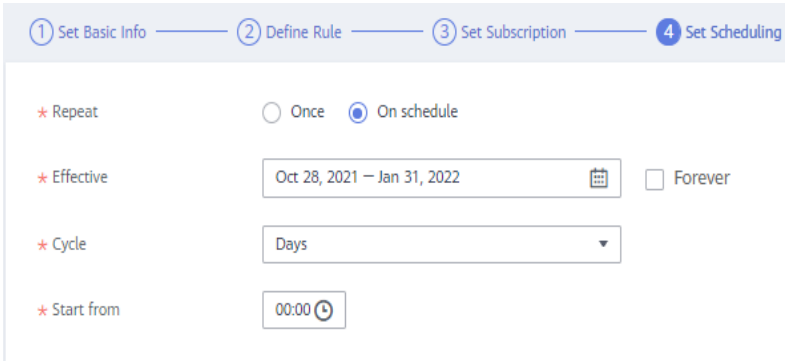
4. Click **Next** and set subscription parameters.



NOTE

If you enable notification, **Alarm triggered** indicates that a notification is sent to the SMN topic when an alarm is generated for the job, and **Run successfully** indicates that a notification is sent to the SMN topic when no alarm is generated for the job.

5. Click **Next** and set scheduling parameters.



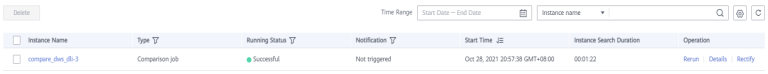
NOTE

Once indicates that the job needs to be manually executed, and **On schedule** indicates that the job is executed automatically based on your configuration. The configuration in the preceding figure indicates that the job is automatically executed every 15 minutes on Oct 27, 2020.

6. Click **Submit**.

Step 3 View the comparison job.

1. In the comparison job list, locate the created job and click **Run** in the **Operation** column.
2. On the displayed **O&M** page, locate the row that contains the comparison job and click **Details** in the **Operation** column to view the running results and logs.

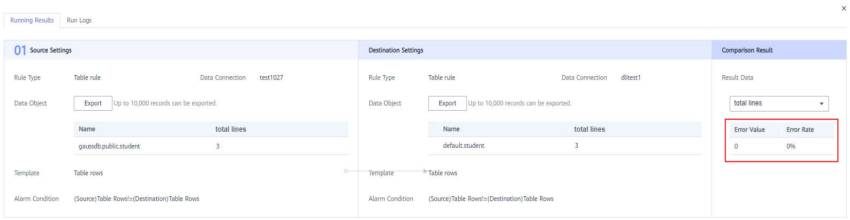


----End

Analyzing the Comparison Result

In the running result, the left pane displays the execution result of the rule for source table rows, and the right pane displays the execution result of the rule for destination table rows.

The error rate indicates the difference between the number of rows of the source and destination tables. If the error rate is 0, the source and destination tables have the same number of rows.



8 DataArts Catalog

This module provides enterprise-class metadata management to clarify information assets. It uses a data map to display a data lineage and panorama of data assets for intelligent data search, operations, and monitoring.

8.1 Data Maps

8.1.1 Overview

Data map facilitates data search and powers data analysis, development, mining, and operations. With data map, you can search for data quickly and make lineage and impact analysis with ease.

- Before data analysis, a data map can be used to search for keywords to narrow down the scope of data to be analyzed.
- A data map can be used to query table details by table names, letting you know how to use a table.
- Through lineage analysis, a data map displays you how a table is generated and where it is applied, and the logic used for processing table fields.

8.1.2 Dashboard

The **Dashboard** page contains two tabs, **Assets** and **Asset Reports**.

- The **Assets** tab page displays information about logical assets, technical assets, and metric assets.
  - Logical assets come from logical entities and data tables defined and released in DataArts Catalog. The number and details of business objects, logical entities, and business attributes are displayed on the **Assets** page.
  - Technical assets come from data connections and metadata collection tasks. The number and details of databases, data tables, and data volumes are displayed on the **Assets** page.
  - Metric assets come from business metrics defined and released in DataArts Architecture. The number and details of business metrics and their details are displayed on the **Assets** page.

- The **Asset Reports** tab page displays logical entities, data tables, asset associations, asset capacities, tags, security levels, top 100 tables by capacity and number of rows, and top 100 buckets by capacity.

Constraints

- Logical assets and metric assets come from DataArts Architecture and are updated if data is synchronized from DataArts Architecture. However, they cannot be deleted directly in DataArts Architecture. Instead, you must locate and delete them in DataArts Catalog.
- Data connections in technical assets come from Management Center and are updated if data is synchronized from Management Center. However, they cannot be deleted directly in Management Center. Instead, you must locate and delete them in DataArts Catalog.
- Information such as databases, tables, and columns in technical assets come from metadata collection tasks. Whether to update and automatically delete such information depends on the parameter settings of metadata collection tasks. For details, see **Task Management**.
- Data lineages in technical assets are updated by job scheduling. Data lineages are generated based on the latest job instances. To delete data lineages, you need to delete jobs or job metadata. Stopping jobs alone does not delete data lineages.

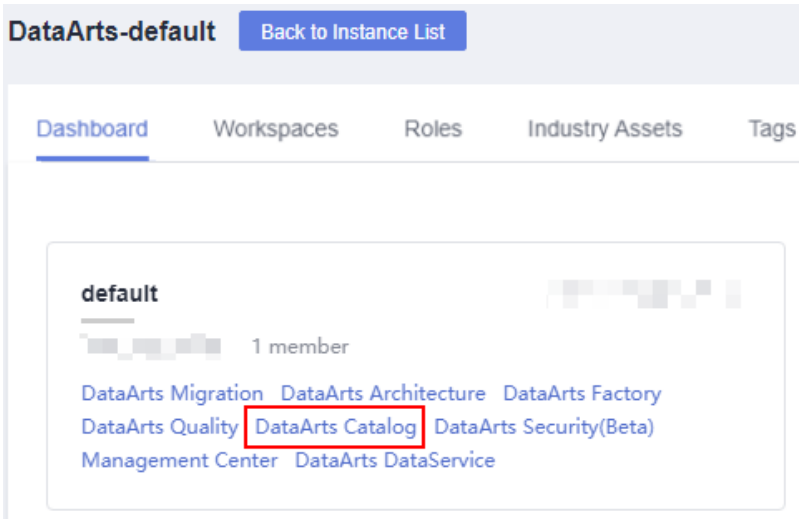
Prerequisites

- Logical entities, data tables, and business metrics have been defined and released in DataArts Architecture.
- A collection task has been created and executed successfully. For details about how to create a collection task, see **Creating a Collection Task**.

Assets

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

Figure 8-1 DataArts Catalog



2. Choose **Data Map** > **Dashboard**.

Figure 8-2 Assets

3. Click **Logical Assets** to view details about logical assets.

   Logical assets come from logical entities and data tables defined and released in DataArts Catalog. The number and details of business objects, logical entities, and business attributes are displayed on the **Assets** page.

4. Click **Technical Assets** to view details about technical assets.

   Technical assets come from data connections and metadata collection tasks. The number and details of databases, data tables, and data volumes are displayed on the **Assets** page.

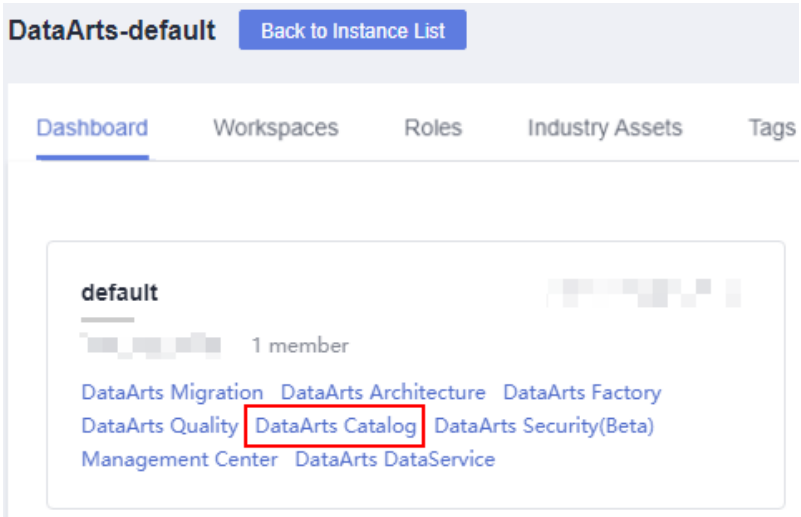
5. Click **Metric Assets** to view details about metric assets.

   Metric assets come from business metrics defined and released in DataArts Architecture. The number and details of business metrics and their details are displayed on the **Assets** page.

Asset Reports

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

Figure 8-3 DataArts Catalog

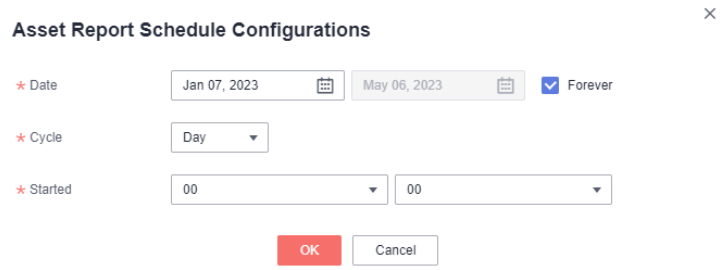


2. Choose **Data Map** > **Dashboard** and click **Asset Reports**.

3. If you access the **Asset Reports** page for the first time, you need to configure asset report tasks. Click in the upper right corner. The **Asset Report Schedule Configurations** dialog box is displayed.

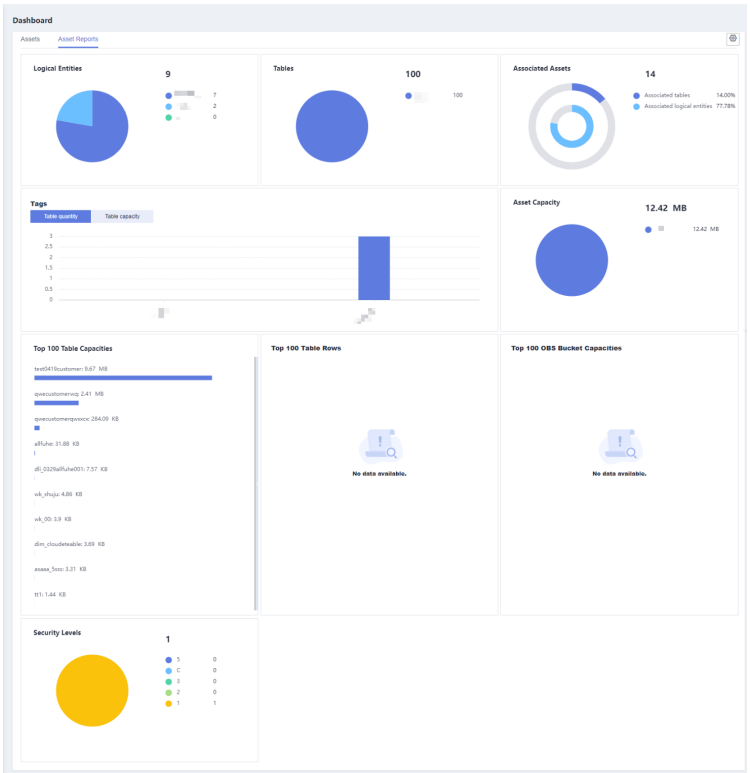
   Set **Date**, **Cycle**, and **Started**. The system will run asset report tasks based on the configuration and update the asset reports.

Figure 8-4 Configuring asset report tasks



4. After the system schedules and runs asset report tasks, you can go to the **Asset Reports** page again to view the logical entities, data tables, asset associations, asset capacities, tags, security levels, top 100 tables by capacity and number of rows, and top 100 buckets by capacity.

Figure 8-5 Asset Reports



8.1.3 Data Catalogs

You can search for and filter assets, and view asset details on the **Data Catalog** page.

- Logical assets come from the logical entities and data tables defined and published in DataArts Architecture.
- Data connections in technical assets come from the data connections in Management Center, and databases, tables, and columns come from metadata collection tasks in DataArts Catalog.
- Metric assets come from the business metrics defined and published in DataArts Architecture.

Constraints

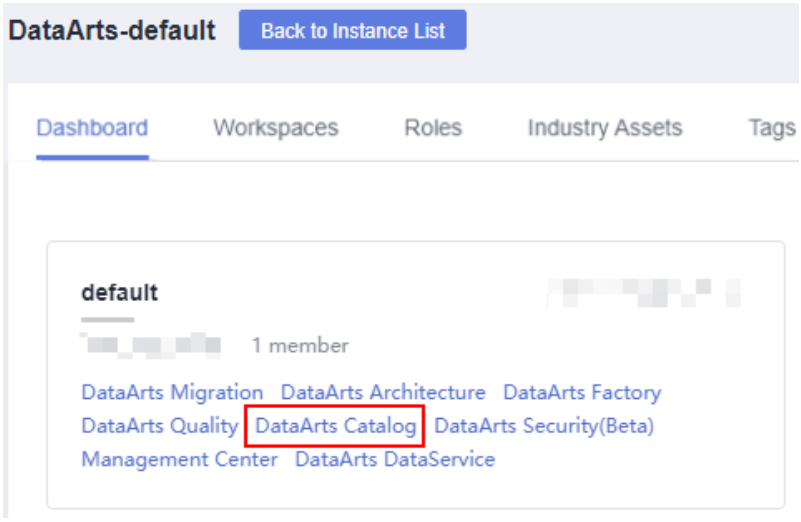
- Logical assets and metric assets come from DataArts Architecture and are updated if data is synchronized from DataArts Architecture. However, they cannot be deleted directly in DataArts Architecture. Instead, you must locate and delete them in DataArts Catalog.
- Data connections in technical assets come from Management Center and are updated if data is synchronized from Management Center. However, they cannot be deleted directly in Management Center. Instead, you must locate and delete them in DataArts Catalog.
- Information such as databases, tables, and columns in technical assets come from metadata collection tasks. Whether to update and automatically delete such information depends on the parameter settings of metadata collection tasks. For details, see **Task Management**.
- Data lineages in technical assets are updated by job scheduling. Data lineages are generated based on the latest job instances. To delete data lineages, you need to delete jobs or job metadata. Stopping jobs alone does not delete data lineages.

Searching for a Data Asset

An asset can be searched by its name, description, or attributes. Fuzzy search is supported.

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

Figure 8-6 DataArts Catalog



2. In the left navigation pane, choose **Data Map** > **Data Catalog**. Click the **Logical Assets**, **Technical Assets**, and **Metric Assets** tabs as needed.
3. In the search box, enter a keyword to search for your desired assets.
   - By their names and description
   - By their attributes, which are displayed on the asset details page

     NOTE

     - You can save the search criteria you set.
     - You can import the search criteria you need.

Filtering an Asset

Technical assets can be filtered by the following criteria:

- Data connection: the data connection that your target asset uses.
- Type: the type of your target asset.
- Classification: the category that your asset is classified into.
- Tag: the tag that your asset includes.
- Security level: the security level of your target asset.

The following uses **type** as an example to demonstrate how to filter an asset.

Step 1 Select **Table** under **Types**. Table assets are displayed.

Step 2 In the **Types** area, **Table**, **Column**, **Database**, **Bucket**, and **ColumnFamily** are supported by default. If you select **All**, the system displays assets of all types.

----End

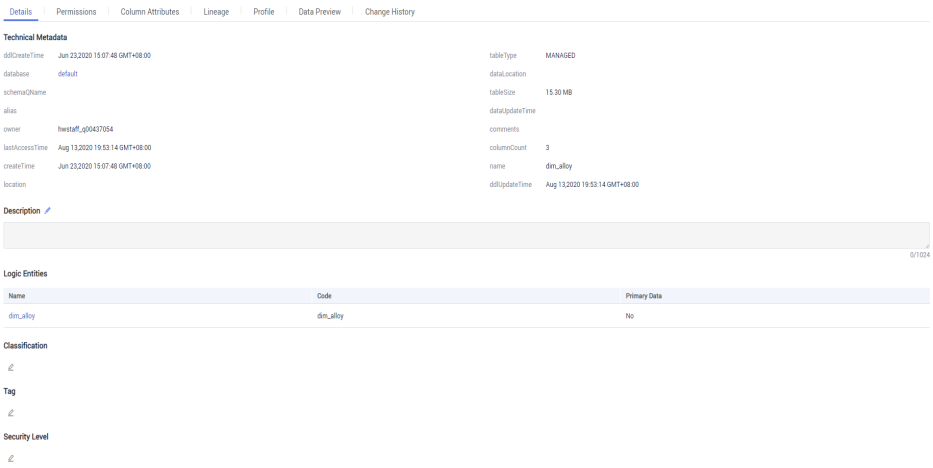
Viewing the Details of an Asset

This section describes how to view data table details on the **Technical Assets** page.

Step 1 In the list of technical assets, select a table and click its name to access its details page.

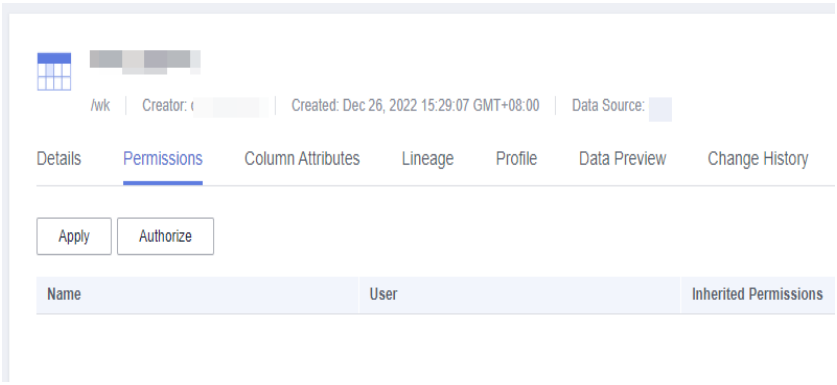
Step 2 On the **Details** tab page, view the basic attributes of the technical metadata; edit the description; add or delete classifications, tags, and security levels for the table, table columns, or OBS objects.

Figure 8-7 Details tab page



Step 3 On the **Permission** tab page, you can apply for data table permissions or grant permissions to other users.

Figure 8-8 Permissions tab page



Step 4 On the **Column Attributes** tab page, view the column attributes of the table; add or delete classifications, tags, and security levels for the data columns; edit the description.

Figure 8-9 Managing column attributes

Step 5 On the **Lineage** tab page, view table lineages and impacts. For details on how to set a data lineage, see **Viewing Data Lineages Through the Data Map**. If a node that supports automatic lineage is configured for a data development job or the

lineage of a node is manually configured, the data lineage can be automatically parsed during job execution and displayed in the data catalog.

Step 6 On the **Profile** tab page, view the profile of the data table. (Currently, this function is available only for GaussDB(DWS), DLI, and OBS data tables. The profile sampling mode is subject to the **metadata collection** task configuration.)

Click **Update** to update the table profile.

Step 7 On the **Data Preview** tab page, preview the business data in the current table. The data can be masked in real time based on the column classification information and the configuration in **Masking Policies**.

- Data assets that use DWS, DLI, MRS Hive, and MySQL data connections can be previewed.
- Column classification information can be automatically set when a collection task is created or manually added in the data classification menu. Automatic classification setting is available only for DWS and DLI data collections.

Step 8 On the **Change History** tab page, view the change history of the table.

----End

8.1.4 Tags

Tags are keywords used to identify the business meaning of data. They help you classify and describe assets for easy search.

Tags can be defined and associated with technical assets for better asset management. For example, you can tag a table as the SDI source data layer or DWI data integration layer.

Tags and Classifications

Tags are highly related keywords that help you classify and describe assets for easy retrieval.

Classification is the process of categorizing assets by category, level, or nature. Classification is top-down. Assets are classified according to certain standards.

The table below lists the differences between tags and classifications.

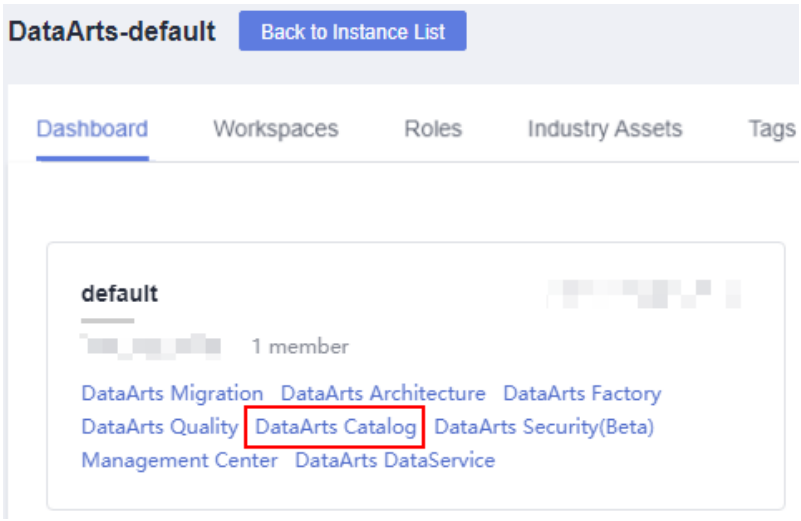
Table 8-1 Differences between tags and classifications

Item	Category	Tag
Exclusiveness	Yes	None
Relationship	Dependent	Relevant (associated)
Creation	Pre-event planning	Any time
Cost	High	Low
Source	For details, see **Data Classifications**.	For details, see **Tags**.

Managing a Tag

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

   Figure 8-10 DataArts Catalog

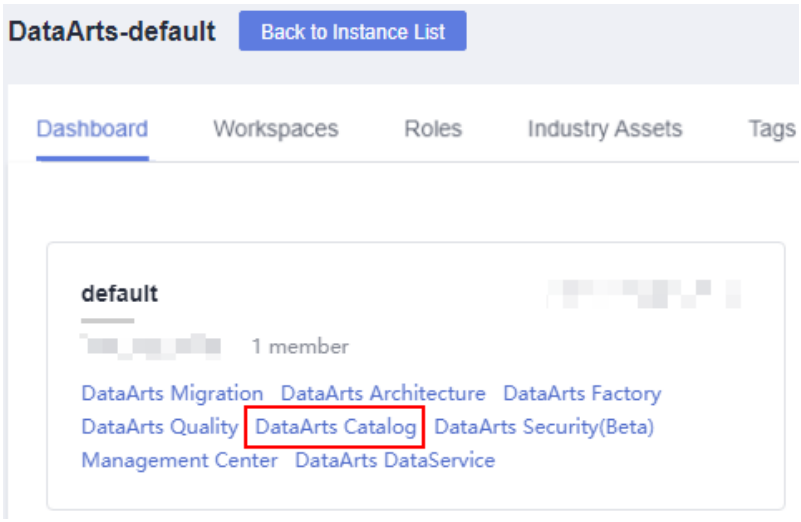


2. Choose **Data Map** > **Tag Management** from the left navigation bar.
3. Click **Create** to create a tag.
   - **Tag Name**: Tag names can include only letters, numbers, and underscores (_). They cannot start with underscores (_) or exceed 100 characters.
   - **Description**: Up to 255 characters are allowed.
4. Select a tag and click **Delete** to delete the tag.
5. Click **Edit** to modify the description of a tag.

Adding a Tag to Identify Data

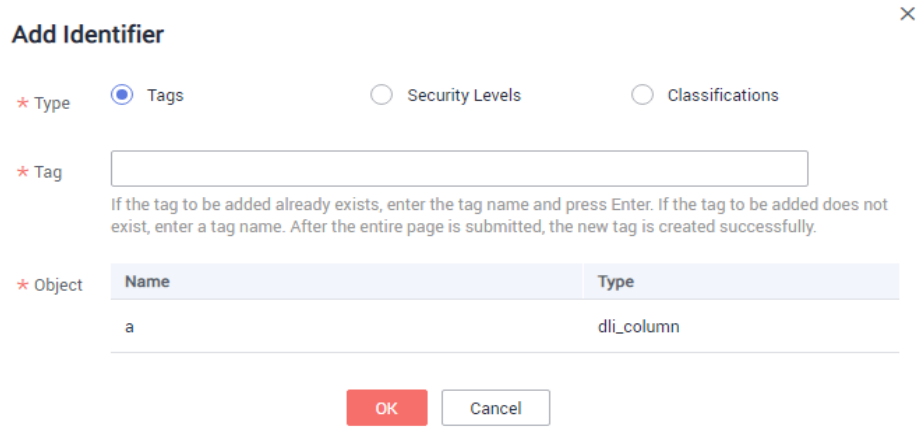
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

   Figure 8-11 DataArts Catalog



2. Choose **Data Map** > **Data Catalog** and click the **Technical Assets** tab.
3. Enter a keyword in the search box, and click the search icon. The search results are listed below the search box.
4. Select the asset that you want to add a tag for and click **Add Identifier** in the upper right corner. In the **Add Identifier** dialog box, select **Tags** for **Type**.

   Figure 8-12 Adding an identifier



5. Set the parameters and click **OK**.

   NOTE

   You can add a new tag or select an existing tag. Existing tags are created by following instructions in **Managing a Tag**.

8.2 Data Permissions

8.2.1 Overview

To ensure data security and controllability, you need to apply for permissions before using data tables. The **Permissions** module facilitates permission control, provides visualized application and approval processes, and supports for permission audit and management. Data is secure and data permission control is convenient.

The **Permissions** module consists of **Data Catalog Permissions**, **Data Table Permissions**, and **Review Center**. The provided functions are:

- Self-service permission application: You can select a data table and quickly apply for the needed permissions online.
- Permission audit: Administrators can quickly and easily view the personnel with the corresponding database table permissions and perform audit management.
- Permission revoking and returning: Administrators can revoke user permissions in a timely manner. Users can also proactively return unnecessary permissions.
- Permission approval and management: Visualized and process-based management and authorization mechanism facilitates post-event tracing.

8.2.2 Data Catalog Permissions

You can manage data catalog permissions.

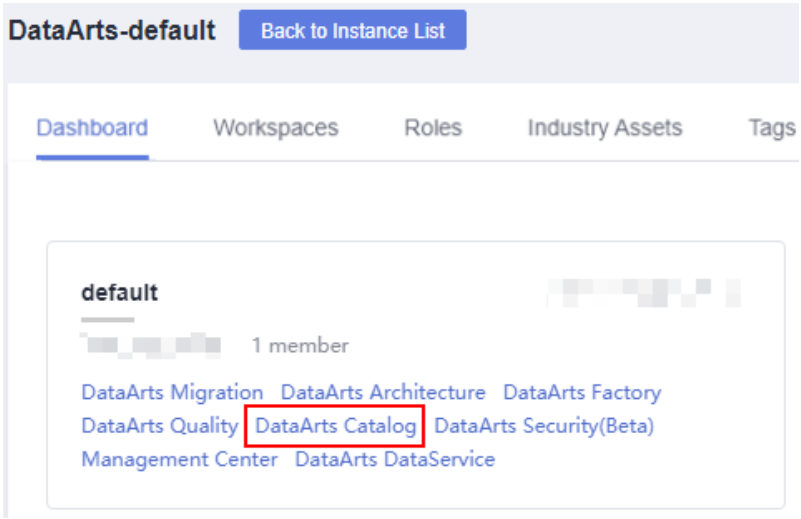
Constraints

- Only workspace admins can create, delete, and modify data catalog permissions rules and set the permissions effective status.
- Workspace developers, operators, and viewers can only view data permissions.

Managing a Data Catalog Permissions Rule

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

   Figure 8-13 DataArts Catalog

2. Choose **Permissions** > **Data Catalog Permissions** from the left navigation bar, and click **Create** on the page displayed to configure a data catalog permissions rule.

   a. **Rule**: Name of a data catalog permissions rule.

   b. **Type**: Currently, only **Tag**, **Security level**, and **Classification** can be used for filtering.

   c. **Scope**: Select available tags, security levels, and classifications.

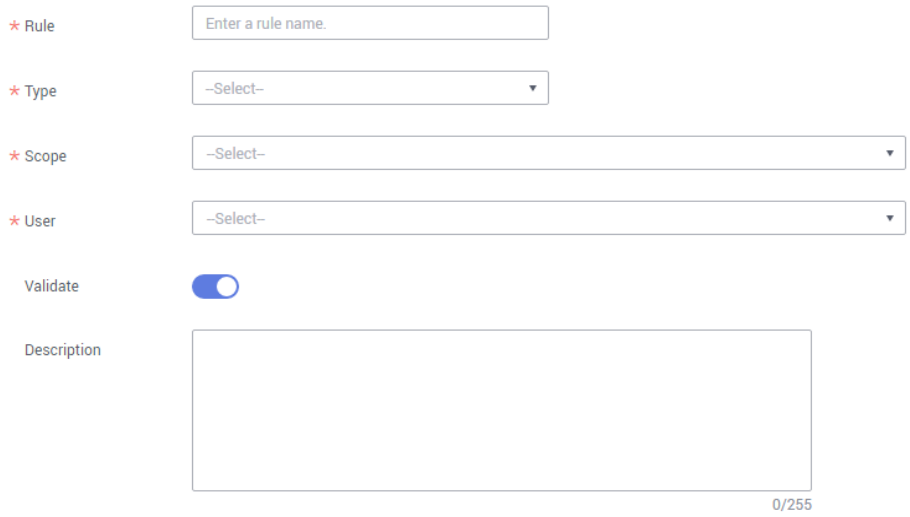
   d. **User**: User to whom the configured data catalog permissions rule applies.

   e. **Validate**: If this function is enabled, the data catalog permissions rule takes effect. Otherwise, the rule does not take effect.

   NOTE

   After a data catalog permissions rule takes effect, only users to whom the configured data directory permissions rule applies can manage data assets with specified tags or classifications. For example, if **Type** is set to **Tag**, **Scope** is set to **test**, and **User** is set to **A**, user A can manage assets with tag **test** after the permissions rule is enabled.

Figure 8-14 Creating a rule



3. In the data catalog permissions rule list, click **Edit** or **Delete** in the **Operation** column to modify or delete the rule.

8.2.3 Table Permissions

On the **My Permissions** page, you can view your table and column permissions in the workspace, and apply for or return the permissions.

Workspace admins have the permissions to manage user permissions. An admin can view the resource permissions of all users in the workspace.

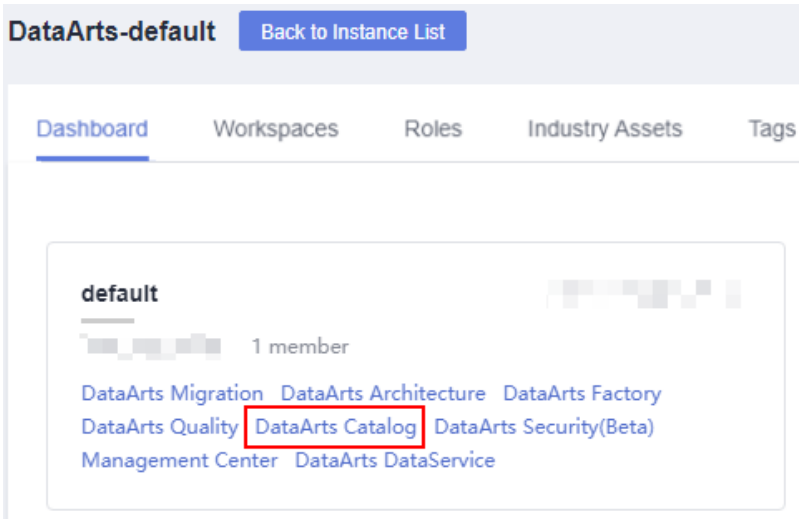
Applying for Table or Column Permissions

NOTE

- The current version supports permissions control only on DLI data tables.
- The table or column permissions you applied for take effect only after being approved by reviewers. Therefore, before applying for the permissions, create a reviewer by referring to **Managing Reviewers**.

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

Figure 8-15 DataArts Catalog

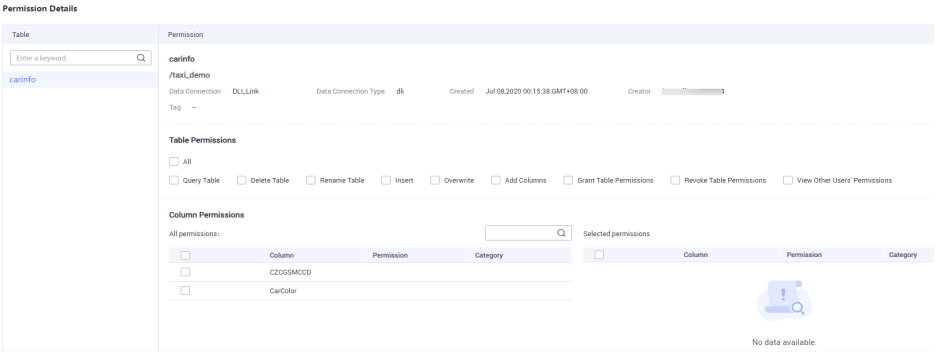


2. Choose **Permissions** > **Data Table Permissions** from the left navigation bar. On the **My Permissions** tab page, click **Apply**.
3. On the page displayed, describe the scenario where the permissions are required, and select the data connection, database, and data table.
4. Select the table or column permissions you want to apply for.
   - Applying for the permissions of a single table or column

     Select the table or column permissions that you do not have but need to use.
   - Applying for the permissions of multiple tables or columns

     After selecting multiple tables, select the table or column permissions to be used in the **Permission Details** area.

Figure 8-16 Applying for permissions on tables and columns



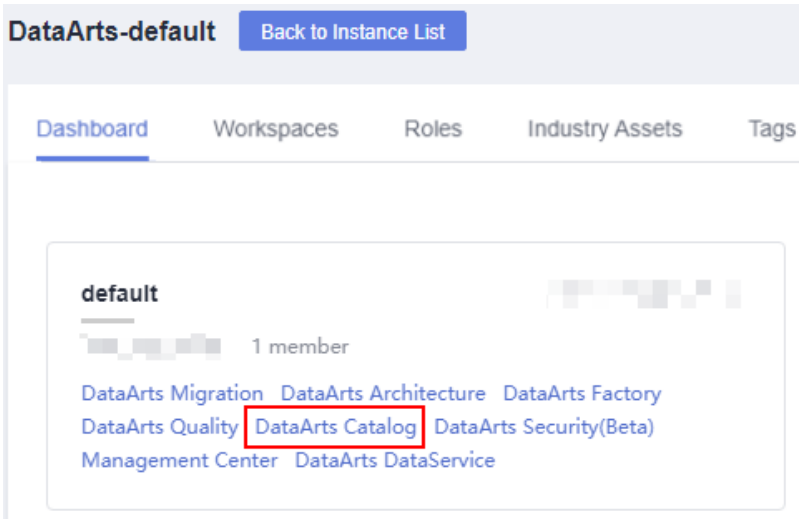
5. Click **OK**. Configure a reviewer and click **OK**.
6. Wait for the reviewer to approve the application. After the application is approved, the permissions take effect.

Managing Existing Table Permissions

You can manage the table or field permissions you already have, including viewing, editing, and returning permissions.

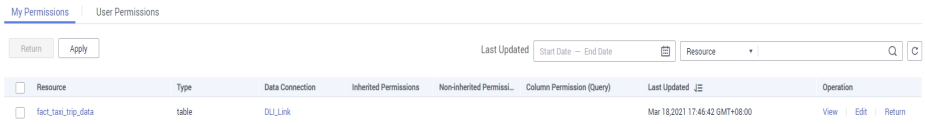
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

   Figure 8-17 DataArts Catalog

2. Choose **Permissions** > **Table Permissions**. On the **My Permissions** tab page, you can perform the following operations:
   - Click **View** in the **Operation** column to view the permissions details.
   - Click **Edit** in the **Operation** column to modify table permissions as needed.
   - Click **Return** in the **Operation** column to return table permissions as needed.

   Figure 8-18 Managing table permissions

Auditing User Permissions

On the **User Permissions** tab page, administrators can view the accounts that have permissions on tables and fields in the same workspace, reclaim the table and field permissions as needed, or grant permissions to users in batches.

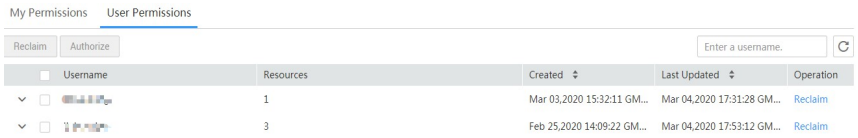
NOTE

Only workspace admins can audit user permissions, including viewing the user list, reclaiming user permissions, or granting permissions to users.

- Viewing accounts with table permissions and the corresponding asset list

  Choose **Data Table Permissions** > **User Permissions** to view the accounts with applied permissions in the same workspace.

Figure 8-19 Viewing accounts with table permissions



- Reclaiming user permissions
  - Choose **Data Table Permissions** > **User Permissions** and click **Reclaim** under **Operation** to the right of the account to reclaim all its permissions.
  - On the **User Permissions** tab page, select the check boxes to the left of one or more usernames, and click **Reclaim** in the upper left corner to revoke their permissions in batches.

Figure 8-20 Reclaiming user permissions

- Granting permissions to users

Figure 8-21 Authorization

- Managing user permissions

  Choose **Data Table Permissions** > **User Permissions**, and click the drop-down arrow to the utmost left of an account to display the assets of the user. Click **View**, **Edit**, and **Return** in the **Operation** column to the right of a specific resource as required.

Figure 8-22 Managing user permissions

8.2.4 Review Center

Constraints

Only workspace admins can manage reviewers, including creating and deleting reviewers.

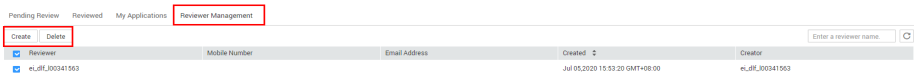
Approval Management

On the **Review Center** page, you can view the application status, applications to be approved, and approved applications, and manage reviewers.

- Reviewer management

  Choose **Permissions** > **Review Center** from the left navigation bar. On the **Reviewer Management** tab page, create and delete reviewers as required. See **Figure 8-23**. The reviewer data refers to the person added in the workspace.

  Figure 8-23 Managing reviewers

- Pending review

  a. Choose **Permissions** > **Review Center** from the left navigation bar, and click the **Pending Review** tab.

     On this page, you can view the applications that need to be approved.

  b. Click **Review** in the **Operation** column to view the application details and approve the application.

  c. After entering the approval comments, approve or reject the application based on the actual situation.

- Reviewed applications

  a. Choose **Permissions** > **Review Center** from the left navigation bar, and click the **Reviewed** tab.

  b. Click **View Details** in the **Operation** column to view the approval records and application content.

- My Applications

  a. Choose **Permissions** > **Review Center** from the left navigation bar, and click the **My Applications** tab.

  b. Click **View Details** in the **Operation** column to view details about an application.

  c. Click **Retry** in the **Operation** column to re-authorize an application.

8.3 Data Security

8.3.1 Overview

Background

Data security provides data lakes with unified data usage protection capabilities throughout the data lifecycle. Sensitive data identification, classification, privacy protection, resource permission control, encrypted data transmission, encrypted storage, data risk identification, and compliance audit help users establish a security warning mechanism and enhance the overall security protection capability, to ensure data security.

Functional Module

Data security includes:

- Data security levels

  You can classify your data into different levels to facilitate data management.
- Data classification rules

  You can classify data to effectively identify sensitive data in databases.
- Masking policies

  Based on the data classification, you can create masking policies to mask data assets and protect privacy.

8.3.2 Data Security Levels

You can manage data security levels, including creating and deleting security levels and adjusting their ranking sequences.

You can create a data classification rule and data masking policy only after you have created a data security level.

Prerequisites

None

Accessing the Data Security Levels Page

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.
2. Choose **DataArts Security** > **Security Levels** from the left navigation bar. On the page displayed, you can create, delete, edit, move up, and move down data security levels as required.
   - Creating a security level: Click **Create** in the upper left corner of the **Data Security Levels** page and enter the name and description.
   - Deleting a security level: Select unnecessary security levels and click **Delete** in the upper left corner of the **Security Levels** page.
   - Adjusting the ranking sequence of a security level: Click **Up** or **Down** to the right of a security level to adjust its sequence.

8.3.3 Data Classifications

You can create data classification rules.

You can create a data masking policy to mask data only after you have created a data classification rule.

Prerequisites

A data security level has been created. For details, see **Data Security Levels**.

Creating a Data Classification Rule

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.
2. Choose **DataArts Security** > **Classifications** from the left navigation bar. On the **Classification Rule** tab page, click **Create**.

   On the page displayed, set the parameters to create a data classification rule. You can either create a rule by using a system template or custom template.

Figure 8-24 Creating a data classification rule

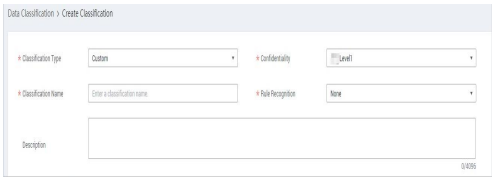


Table 8-2 Parameters for creating a data classification rule

Parameter	Description
Classification Type	The category to which a rule belongs. You can either create a rule by using a system template or custom template.
Confidentiality	Classify the configured data into different levels. If the existing confidentiality does not meet the requirements, go to the confidentiality management page to set security levels. For details, see **Data Security Levels**.
Classification Template	This parameter is available when **Classification Type** is set to **Built-in**. You can select a system sensitive data identification template based on service requirements, for example, **Time**, **Mobile number**, and **License plate number**.
Classification Name	• If **Classification Type** is set to **Built-in**, a classification name is automatically generated based on the classification template selected. • If **Classification Type** is set to **Custom**, you can customize a classification name. **NOTE** The name of a data classification rule must be unique.
Rule Recognition	This parameter is available when **Classification Type** is set to **Custom**. Regular expressions are supported.
Regular Expression	• **Content recognition**: You can customize a regular expression. • **Column name recognition**: Both exact match and fuzzy match are supported. Multiple fields can be matched.
Description	A description of the data classification rule to create.

Creating a Group

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.
2. Choose **DataArts Security** > **Classifications** from the left navigation bar. On the **Groups** tab page, click **Create**.

   In the **Create Group** dialog box, set the parameters and click **OK**.

   Set the parameters by referring to **Table 8-3** and select classification rules in the list.

   The selected rules are displayed in the list on the right.

   Table 8-3 Parameters for creating a group

Parameter	Description
Name	The name of a group. Only letters, numbers, and underscores (_) are allowed.
Description	Information to better identify the group. It cannot exceed 4,096 characters.

8.3.4 Masking Policies

You can create a data masking policy and perform masking query in DataArts Catalog.

Prerequisites

- A data classification rule has been created. For details on how to create a classification rule, see **Data Classifications**.
- A data connection and a data table have been created, and sensitive data has been collected by DataArts Catalog.

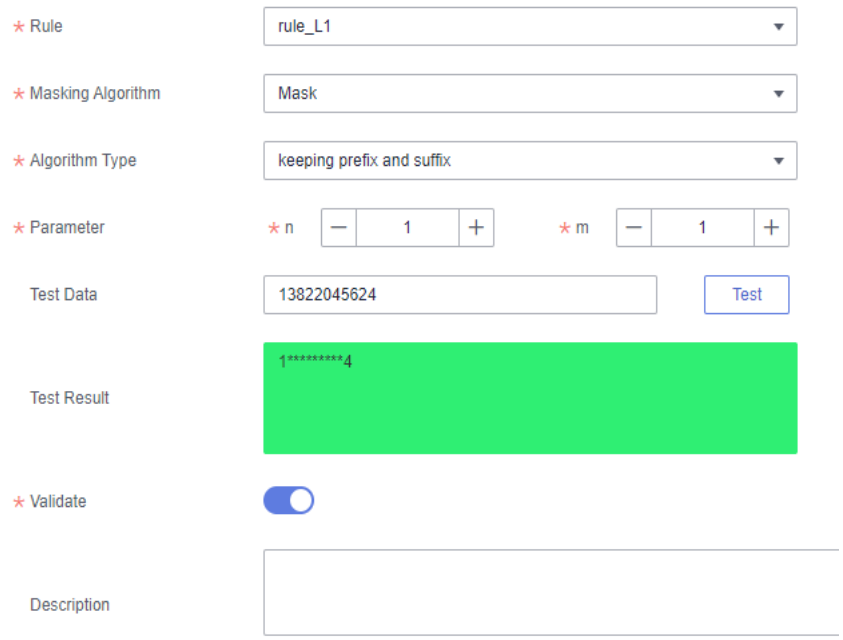
Creating a Masking Policy

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.
2. Choose **DataArts Security** > **Masking Policies** from the left navigation bar, and click **Create** on the page displayed.
3. Set **Classification Rule**, **Masking Algorithm**, and **Algorithm Type**. The options for **Masking Algorithm** include **Mask**, **Truncate**, and **Hash**. Each masking algorithm has multiple algorithm types. Select an algorithm type as required. After the configuration, click **OK**.

   NOTE

   A data classification rule can be bound to only one masking algorithm.

Figure 8-25 Creating a masking policy



4. After you configured the making algorithm, you can perform an online test. Enter the test data, and click **Test**. You can verify the result in the **Test Result** text box.
5. Enable or disable **Status**. The masking policy takes effect only when **Status** is enabled.

Viewing the Data Masking Effect

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.
2. Choose **Data Map** > **Catalog** from the left navigation bar.
3. In a list of asset results, click a table name to access its details page.
4. Click **Data Preview** to view the data masking effect.

8.4 Metadata Collection

8.4.1 Overview

Metadata is data about data. Metadata streamlines source data, data warehouses, and data applications, and records the entire process from data generation to data consumption. Metadata mainly refers to model definitions in the data warehouse and mappings between layers. It also describes the monitoring data status of the data warehouse and running status of ETL tasks. In the data warehouse system, metadata helps data warehouse administrators and developers easily locate the data they are looking for, improving the efficiency of data management and development.

In DataArts Studio, metadata may be used to describe the attributes of data (such as the data connection, type, name, and size) or other related information of data (such as the data owner, tag, category, and security level).

Metadata is classified into technical metadata and business metadata by function.

- Technical metadata is data that stores technical details of a data warehouse system and is used to develop and manage data warehouses. In DataArts Studio, technical metadata is technical assets, including databases, data tables, and data volume and their details.
- Business metadata describes data in a data warehouse from the business perspective. It provides a semantic layer between users and actual systems, enabling business personnel who do not understand computer technologies to understand data in the data warehouse. In DataArts Studio, business metadata includes logical assets and metric assets. Business assets include business objects, logical entities, and business attributes and their details. Metric assets include business metrics and their details.

Technical metadata in DataArts Studio are obtained through metadata collection tasks. You can view metadata on the **Data Map** page only after you have created and run a metadata collection task.

8.4.2 Task Management

You can create collection tasks by configuring metadata collection policies. Different types of data sources require different collection policies. Metadata management allows you to collect technical metadata using the configured collection policies.

Constraints

- If the collection scope is not specified for a metadata collection task, all data tables and files of a data connection are collected by default. After the collection task is complete, if data tables or files are added to the data connection, you must run the metadata collection task again to collect the new data tables or files.
- Before collecting Oracle metadata, ensure that the database user of the data connection has the permission to read and write data tables and read metadata. For details, see how to assign permissions to users in **Configuring an Oracle Connection**.

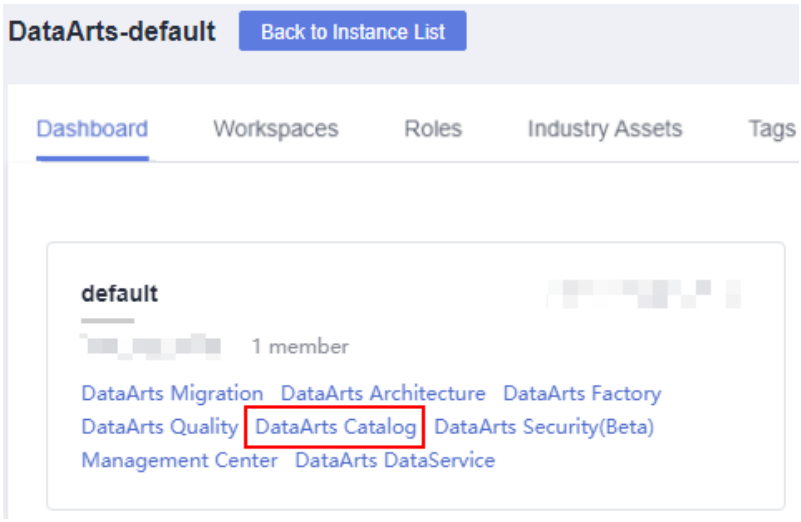
Prerequisites

- Metadata of the following types of data sources can be collected: DWS, DLI, MRS HBase, MRS Hive, RDS (MySQL), RDS (PostgreSQL), and Oracle. To obtain metadata, you must first create data connections in Management Center. To collect metadata from other data sources (such as OBS, CSS, and GES), you do not need to create data connections in Management Center.
- Before you can collect the metadata of Hudi tables by collecting the MRS Hive metadata, you must enable synchronization of the Hive table configuration for Hudi tables.

Creating a Collection Task

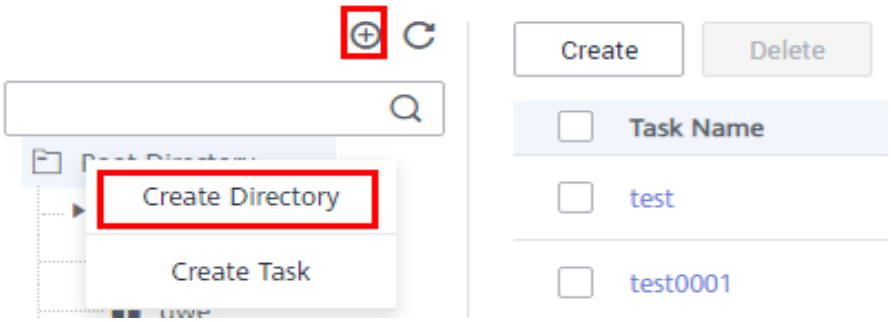
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

   Figure 8-26 DataArts Catalog

2. Choose **Metadata Collection** > **Task Management** from the left navigation bar.
3. Select the directory for the collection task. If no directory is available, create one as **Figure 8-27** shows.

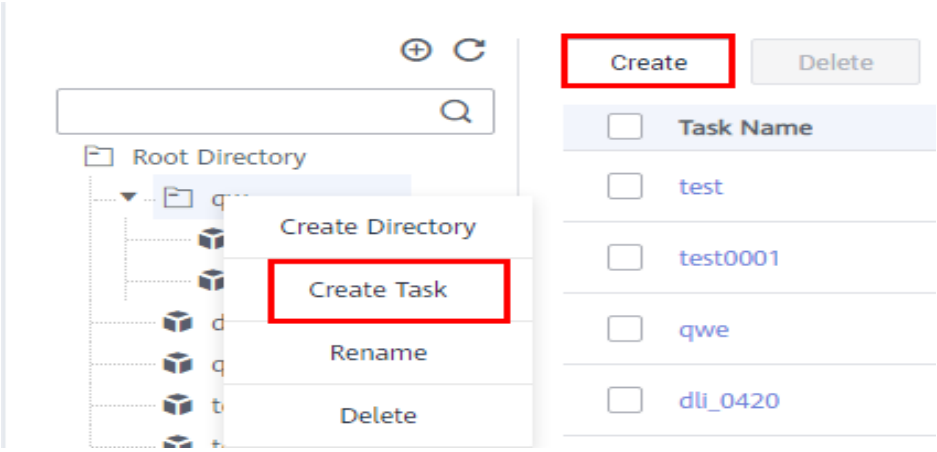
   Figure 8-27 Directory that stores the collection task to create

4. Click **Create** in the upper part of the displayed page or right-click **Task name** and choose **Add Task** from the shortcut menu. On the page displayed, set the parameters.

   Figure 8-28 shows the entries for creating a task.

Figure 8-28 Entries for creating a collection task



a. Set the basic configuration based on **Table 8-4**.

Table 8-4 Basic configuration parameters

Parameter	Description
Task Name	Name of a collection task. The value can contain only letters, numbers, and underscores (_), and cannot exceed 62 characters.
Description	Information to better identify the collection task. Length of the description cannot exceed 255 characters.
Select Directory	The directory that stores the collection task. You can select an existing one. **Figure 8-27** shows the directory.

b. Configure data source information based on **Table 8-5**.

Table 8-5 Data source parameters

Parameter	Description
Data Connection Type	Select a data connection type from the drop-down list box. **NOTE** Metadata of the following types of data sources can be collected: DWS, DLI, MRS HBase, MRS Hive, RDS (MySQL), RDS (PostgreSQL), and Oracle. To obtain metadata, you must first create data connections in Management Center. To collect metadata from other data sources (such as OBS, CSS, and GES), you do not need to create data connections in Management Center.

Parameter		Description
• **DWS** • **DLI** • **MRS HBase** • **MRS Hive** • **ORACLE** • **RDS**	Data Connection Name	• To use an existing data connection, select a value from the drop-down list. • To use a data connection that does not exist, click **Create** to add one.
	Database (or **Database and Schema** and **Namespace**)	Database, schema, or namespace and data table from which data will be collected • Click **Set** next to **Database** (or **Database and Schema** or **Namespace**) to set the range of databases (or databases and schemas or namespaces) to be scanned by the collection task. If this parameter is not set, all databases (or databases and schemas or namespaces) under the data connection are scanned by default. • Click **Set** next to **Table** to set the range of tables to be scanned by the collection task. If this parameter is not set, all tables in the database (or database and schema or namespace) are scanned by default. • If neither the database (or database and schema or namespace) nor the data table is set, the task scans all data tables of the selected data connection. • Click **Clear** to delete the selected database (or database and schema or namespace) and data table.
	Table	
CSS	Cluster	Select the CSS cluster for storing the data to be collected. You can also click **Create** to create a CSS cluster. After the CSS cluster is created, click **Refresh** and select the new CSS cluster.
	CDM Cluster	Select the agent provided by the CDM cluster. You can also click **Create** to create an agent. After the agent is created, click **Refresh** and select the new agent.
	Index	Index, similar to "database" in the relational database (RDB), stores Elasticsearch data. It is a logical space that consists of one or more shards.
GES	Graph	Select graphs that store structured data based on "relationships".
	CDM Cluster	Select the agent provided by the CDM cluster. You can also click **Create** to create an agent. After the agent is created, click **Refresh** and select the new agent.

Parameter		Description
OBS	OBS Bucket	Select the OBS bucket from which data will be collected.
	OBS Path	Select the path of the OBS bucket from which data will be collected.
	Collection Scope	Select the range of data to be collected. • If you select **This folder**, the collection task collects only the objects in the folder set in the OBS path. • If you select **This folder and subfolders**, the collection task collects all objects in the folder set in the OBS path, including the objects in the sub-folders.
	Collected Content	Select the content of data to be collected. • If you select **Folders and objects**, the collection task collects folders and objects. • If you select **Folders**, the collection task collects only folders.
DIS	Collect Dump Task	If **Yes** is selected, the dump task is collected.
	Collection Channel	A DIS instance is a stream. This parameter is used to specify a stream used for data collection.

c. Set parameters under **Metadata Collection**. See **Table 8-6**.

NOTE

Metadata collection parameters are available only for DWS, DLI, MRS HBase, MRS Hive, RDS, or Oracle connections.

Table 8-6 Parameters for metadata collection

Parameter	Description
The data source metadata has been updated.	When metadata in a data connection changes, you can configure an update policy to set the metadata update mode in the data catalog. Note that the configured update and deletion policies apply only to the databases and data tables configured by yourself. • If you select **Update metadata in the data directory only**, the collection task updates only the metadata that has been collected in the data catalog. • If you select **Add new metadata to the data directory only**, the collection task collects only metadata that exists in the data source but does not exist in the data catalog. • If you select **Update metadata in the data directory and add metadata**, the collection task fully synchronizes metadata from the data source. • If you select **Ignore the update and addition operations**, the metadata in the data source is not collected.
The data source metadata has been deleted.	When metadata in a data connection changes, you can configure a deletion policy to set the metadata update mode in the data catalog. • If you select **Delete metadata from data directory**, when some metadata in the data source is deleted, the corresponding metadata is also deleted from the data catalog. • If you select **Ignore the deletion**, when some metadata in the data source is deleted, the corresponding metadata is not deleted from the data catalog.

d. Set parameters when **Data Summary** is selected. See **Table 8-7** for details.

NOTE

- **Data Summary** parameters are available only for DWS, DLI connections.
- You are advised not to select **Data Summary** unless necessary. Selecting this option will increase the SQL execution workload. As a result, the metadata collection task may take a longer time than expected.

Table 8-7 Parameters

Parameter	Description
Full data	If this option is selected, a data profile is generated in the data catalog based on all data collected. This mode applies to scenarios where the data volume is less than 1 million.
Sampled data, first *x* rows	If this option is selected, a data profile is generated in the data catalog based on all data collected. This mode is applicable to scenarios with a large amount of data.
Randomly collect *x*% records of data from all data	If this option is selected, a data profile is generated in the data catalog based on all data collected. This mode is applicable to scenarios with a large amount of data.
Data Lake Insight Queue	The queue used to obtain profile data and execute DLI SQL statements. If you select **Collect unique value**, the number of unique values in the collected table is calculated and displayed on the **Profile** tab page in the data catalog.

e. Set parameters when **Data Classification** is selected. (This option is available only when DataArts Catalog provides data security functions. The data classification cannot be associated with a sensitive data identification rule created in the independent DataArts Security module.)

- If you select **Data Classification** and create a classification rule group or select an existing classification rule group by referring to **Data Classifications**, data will be automatically identified and a classification will be added.
- If you select **Update the data table security level based on the data classification result**, the table security level must be the same as the highest security level of the matched classification rules.
- If you select **Manually** for **Synchronize Data**, classification rules and security levels are not automatically added to **Column Attributes** of **Data Catalog** under **Data Map**. Go to the **Task Monitoring** page. Locate the target instance and choose **More** > **View Scanning Result** to view the execution result of the collection task and check whether the classification result matches. Select the check box of the classification matching field and click **Synchronize** to manually synchronize the classification rule and security level.

NOTE

Only when you choose the DWS or DLI data source, you can add data classifications for automatic data identification. In addition, you can add classification rules only for columns in the data tables and OBS objects.

5. Click **Next** and select a scheduling mode.

   Once: If the execution duration of a task exceeds the configured timeout duration, the task is considered failed.

   Repeating: See **Table 8-8** for details.

   NOTE

   1. If **Once** is selected, a manual task instance is generated. A manual task has no dependency on scheduling and must be manually triggered.
   2. If **Repeating** is selected, a periodic instance is generated. A periodic instance is an instance snapshot that is automatically scheduled when the scheduled execution time is arrived.
   3. When a periodic task is scheduled once, an instance workflow is generated. You can perform routine O&M on scheduled instance tasks, such as viewing the running status, stopping and rerunning the scheduled tasks.

   Table 8-8 Parameters

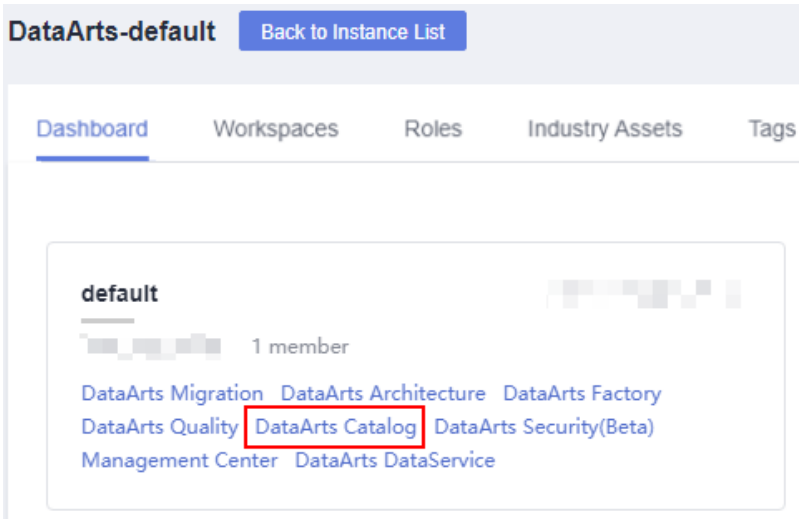
Parameter	Description
Scheduling Date	The period during which a scheduling task takes effect.
Scheduling Cycle	The frequency at which the scheduling task is executed, which can be: • Minutes • Hours • Days • Weeks
Start Time	Start time of periodic scheduling, which is used together with the start time in **Scheduling Date**.
Time Interval	Interval between two periodic scheduling operations A scheduling task instance starts even if the previous scheduling task instance has not ended. A collection task supports concurrent running of multiple instances.
End Time	End time of periodic scheduling, which is used together with the end time in **Scheduling Date**.
Timeout	Timeout duration for a task instance. If a task runs longer than the value of this parameter, the task fails to be executed.
Start	If this check box is selected, the task is scheduled immediately.

6. Click **Submit**. The collection task is created.

Managing a Collection Task

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

   Figure 8-29 DataArts Catalog

2. Choose **Metadata Collection** > **Task Management** from the left navigation bar.

Then, you can view all created collection tasks.

Table 8-9 Parameters for managing collection tasks

Parameter	Description
Task Name	The name of a collection task. Click a collection task name to view the collection policies and scheduling properties.
Type	The name of a data connection.
Scheduling Status	The scheduling status of a collection task. You can click to view only tasks of the specified statuses.
Scheduling Cycle	The scheduling frequency of a collection task. You can click to view only tasks of the specified frequencies.
Description	The description of a collection task.
Creator	The creator of a collection task.
Last Executed On	The last time when the collection task ran.

Parameter	Description
Operation	You can perform the following operations on a created collection task: • **Edit**: Modify the parameters that are closely related to the policies of collection tasks whose status is **Started**, **Not started**, or **Failed**. The data source type cannot be modified. • **Run**: Click **Run** to run a collection task once and view its status and related logs on the **Task Monitoring** page. • **Start Scheduling**: If the status of a task is **Stopped**, you can start scheduling the task based on the configured scheduling mode. • **Stop Scheduling**: When the scheduling status is **Scheduling**, you can stop the scheduling.

8.4.3 Task Monitoring

You can monitor the running status of metadata collection tasks, view collection logs, and perform operations such as rerunning collection tasks.

On the **DataArts Catalog** page, choose **Metadata Collection** > **Task Monitoring** in the left navigation pane. On the page displayed, monitor the created collection tasks. See **Table 8-10** for details.

Table 8-10 Parameters for monitoring a collection task

Parameter	Description
Task Name	The name of a collection task.
Instance Status	The status of an instance (collection task), which can be: • Successful • Partially successful • Executing • Failed • Running exception • Paused: Task monitoring is paused due to management plane upgrade. After the upgrade is complete, the monitoring will recover.
Schedule	The scheduling mode of the collection task. The options are **Schedule once** and **Schedule periodically**.
Time Interval	The scheduling period of the collection task.
Start Time	The time when the collection task restarts running.
End Time	The time when the collection task stops running.

Parameter	Description
Running Duration (min)	The duration that the collection task has run.
Operation	The operations that can be performed on the collection task under monitoring: • **Rerun**: Instances whose statuses are **Failed** or **Succeeded** can be rerun. • **View Log**: You can view instance logs. **NOTE** Click **View Log** to view the run logs of metadata collection, data summary, and data classification tasks in real time. • **More > Cancel**: You can perform this operation only when **Manually** is selected for **Synchronize Data** under **Data Classification** during the creation of the collection task. Instances whose statuses are **Executing** can be stopped. • **More > View Scanning Result**: You can perform this operation only when **Manually** is selected for **Synchronize Data** under **Data Classification** during the creation of the collection task. You can view the execution result of the collection task instance to check whether the classification result is matched. Select the check box of the classification matching field and click **Synchronize** to manually synchronize the classification rule and security level.

8.5 Tutorials

8.5.1 Developing an Incremental Metadata Collection Task

Configuring and running a collection task is the prerequisite for building data assets. This section describes how to create different types of metadata collection tasks.

Scenario 1: Adding Metadata Only

Create a collection task to collect new tables only.

For example, if table4 is newly added:

- Data table metadata before collection: table1, table2, table3
- Data table metadata after collection: table1, table2, table3, **table4**

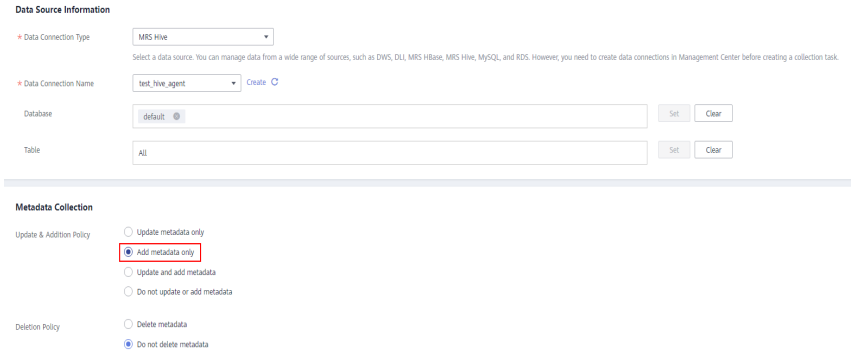
If you only want to collect table 4 in the preceding figure, perform the following steps (on condition that table 1, 2, and 3 are already in DataArts Catalog):

Step 1 Access the **DataArts Catalog** module on the DataArts Studio console.

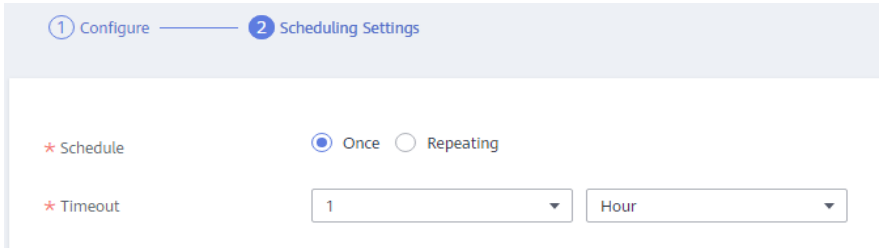
Step 2 In the navigation pane on the left, choose **Collection Tasks**.

Step 3 Click **Create**.

Step 4 Configure parameters for the task.



Step 5 Click **Next** and set scheduling parameters.



Step 6 Click **Submit** to create a collection task.

Step 7 In the task list, locate the created task and click **Run** or **Start Schedule** in the **Operation** column to go to the **Task Monitoring** page and view the task status.

----End

Scenario 2: Updating Existing Metadata and Adding New Metadata

Create a collection task to collect all tables, including existing and new ones.

For example, if table4 is newly added:

- Data table metadata before collection: table1, table2, table3
- Data table metadata after collection: **table1**, **table2**, **table3**, **table4**

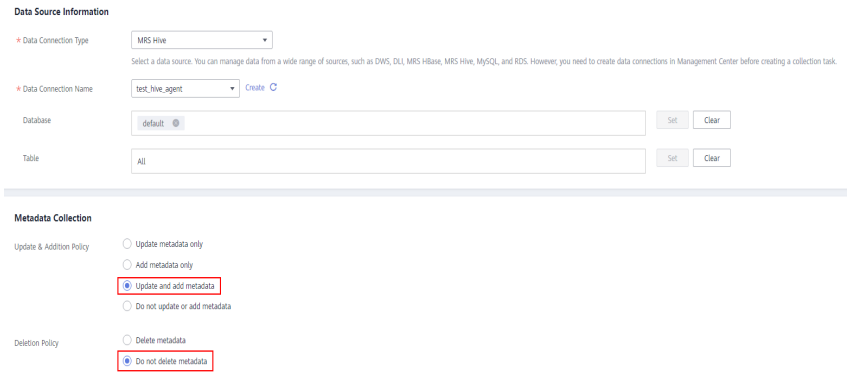
If you want to collect all tables in the preceding figure, perform the following steps:

Step 1 Access the **DataArts Catalog** module on the DataArts Studio console.

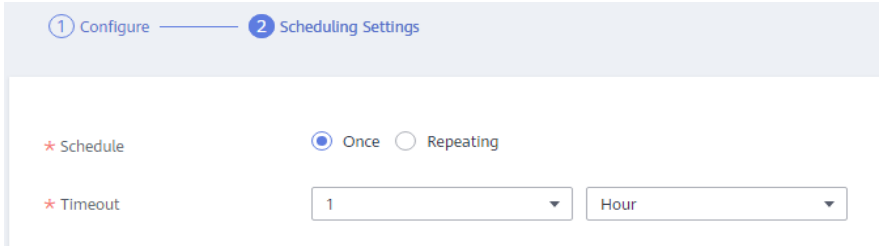
Step 2 In the navigation pane on the left, choose **Collection Tasks**.

Step 3 Click **Create**.

Step 4 Configure parameters for the task.



Step 5 Click **Next** and set scheduling parameters.



Step 6 Click **Submit** to create a collection task.

Step 7 In the task list, locate the created task and click **Run** or **Start Schedule** in the **Operation** column to go to the **Task Monitoring** page and view the task status.

----End

Scenario 3: Updating Existing Metadata Only

Create a collection task to collect existing tables.

For example, if table4 is newly added:

- Data table metadata before collection: table1, table2, table3
- Data table metadata after collection: **table1**, **table2**, **table3**

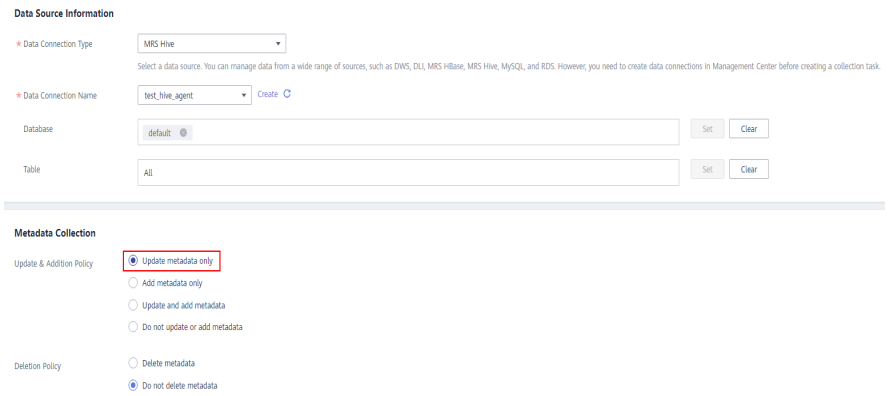
If you want to collect table 1, 2, and 3 in the preceding figure, perform the following steps:

Step 1 Access the **DataArts Catalog** module on the DataArts Studio console.

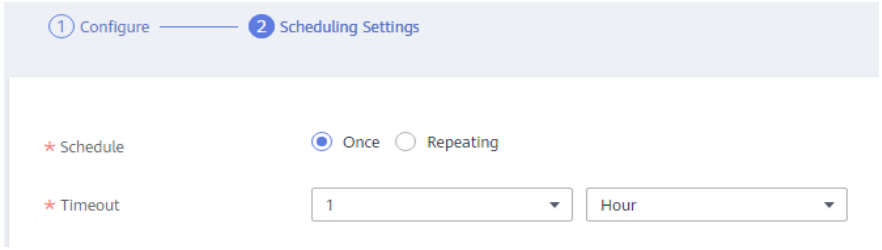
Step 2 In the navigation pane on the left, choose **Collection Tasks**.

Step 3 Click **Create**.

Step 4 Configure parameters for the task.



Step 5 Click **Next** and set scheduling parameters.



Step 6 Click **Submit** to create a collection task.

Step 7 In the task list, locate the created task and click **Run** or **Start Schedule** in the **Operation** column to go to the **Task Monitoring** page and view the task status.

----End

Scenario 4: Updating and Deleting Existing Metadata and Adding New Metadata

Create a collection task to delete existing tables.

For example, if table1 is deleted from the database:

- Data table metadata before collection: table1, table2, table3
- Data table metadata after collection: **table2**, **table3**

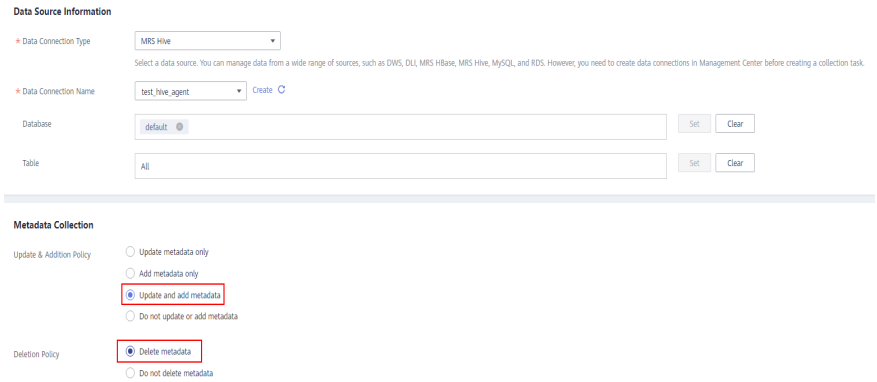
If you want to delete table1, perform the following steps:

Step 1 Access the **DataArts Catalog** module on the DataArts Studio console.

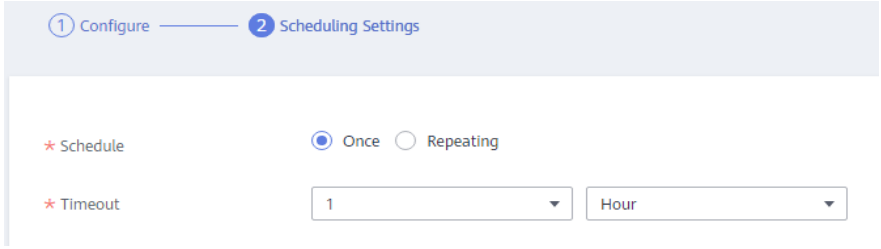
Step 2 In the navigation pane on the left, choose **Collection Tasks**.

Step 3 Click **Create**.

Step 4 Configure parameters for the task.



Step 5 Click **Next** and set scheduling parameters.



Step 6 Click **Submit** to create a collection task.

Step 7 In the task list, locate the created task and click **Run** or **Start Schedule** in the **Operation** column to go to the **Task Monitoring** page and view the task status.

----End

8.5.2 Viewing Data Lineages Through the Data Map

8.5.2.1 Overview

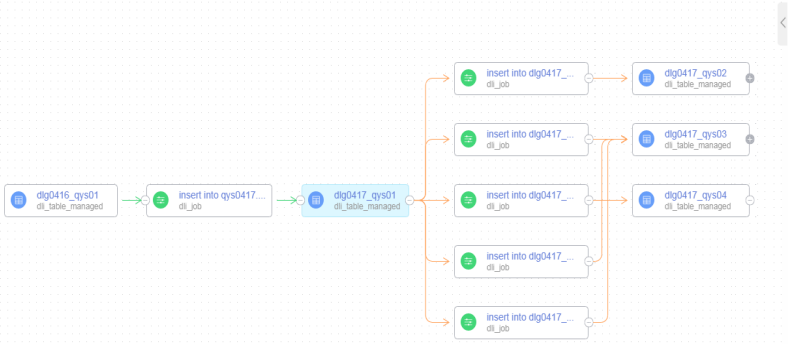
What Is Data Lineage?

In the era of big data, various types of data are rapidly generated due to explosive data growth. The massive and complex data information is converged, transformed, and transferred to generate new data and aggregate into an ocean of data.

During this process, a relationship is formed between the data, and these relationships are their lineages. They are analogous to the genetic relationships between people. However, in contrast from our human lineages, data lineages have the following distinct features:

- **Belongingness**: Specific data belongs to a specific organization or individual.
- **Multi-source**: One piece of data can have multiple sources. One piece of data may be generated by processing multiple pieces of data, and there may be multiple such processes.
- **Traceability**: The data lineage is traceable. It reflects the data lifecycle and the entire process from data generation to data disappearance.
- **Hierarchy**: The data lineage is hierarchical. Data classification and summary form new data, and different levels of description result in data layers.

Figure 8-30 Data lineage example



How DataArts Studio Data Lineage Is Implemented

- Generation of data lineages:

  The DataArts Studio data lineage parsing solution supports automatic lineage analysis and manual lineage configuration. Automatic lineage parsing is recommended. In this mode, lineages can be generated without manual configuration. If automatic lineage parsing is not supported, manually configure lineages.

  - Automatic lineage parsing: Lineages are automatically generated after the system parses the data processing and data migration nodes in data development jobs. No manual configuration is required. For details about the node types and scenarios that support automatic lineage parsing, see **Automatic Lineage Parsing**.
  - Manual lineage configuration: Customize the input and output tables of lineages in data development job nodes. If you configure lineages manually for a node, the automatic lineage parsing does not take effect for this node. For details about the node types that support manual lineage configuration, see **Manually Configuring a Lineage**.

- Display of data lineages:

  You need to create a metadata collection task in DataArts Catalog first. When a data development job meets the **automatic lineage parsing requirements** or **lineages have been manually configured**, and when the job is successfully scheduled, you can view the data lineages in DataArts Catalog.

8.5.2.2 Configuring Data Lineages

The DataArts Studio data lineage parsing solution supports automatic lineage analysis and manual lineage configuration. Automatic lineage parsing is recommended. In this mode, lineages can be generated without manual configuration. If automatic lineage parsing is not supported, manually configure lineages.

- Automatic lineage parsing: Lineages are automatically generated after the system parses the data processing and data migration nodes in data development jobs. No manual configuration is required. For details about the node types and scenarios that support automatic lineage parsing, see **Automatic Lineage Parsing**.
- Manual lineage configuration: Customize the input and output tables of lineages in data development job nodes. If you configure lineages manually

for a node, the automatic lineage parsing does not take effect for this node. For details about the node types that support manual lineage configuration, see **Manually Configuring a Lineage**.

Constraints

Currently, field-level lineage parsing is not supported.

Automatic Lineage Parsing

Automatic lineage parsing does not require manual configuration. When a data development job contains the nodes and scenarios listed in **Table 8-11**, the system can automatically parse lineages.

NOTE

The lineage of an SQL node can be parsed using multiple SQL statements, and column-level lineage parsing is supported. A single SQL statement cannot contain semicolons (;).

Table 8-11 Job nodes and scenarios that support automatic lineage parsing

Job Node	Supported Scenario
DLI SQL	• Lineages generated by data insertion between DLI tables • Lineages between OBS files generated by table creation statements and DLI tables
DWS SQL	Lineages between DWS tables generated by DML operations such as "Insert into"
MRS Hive SQL	Lineages between MRS tables generated by DML operations such as "Insert into/overwrite"
MRS Spark SQL	Lineages between MRS tables generated by DML operations such as "Insert into/overwrite"
CDM Job	Lineages generated during table file migration between MRS Hive, DLI, RDS, CSS, DWS, and OBS
ETL Job	Data lineages generated by ETL tasks between DLI, OBS, MySQL, and DWS.

Manually Configuring a Lineage

In a DataArts Studio data development job, you can customize the input and output tables of lineages on the nodes of the job. If you configure lineages manually for a node, the automatic lineage parsing does not take effect for this node.

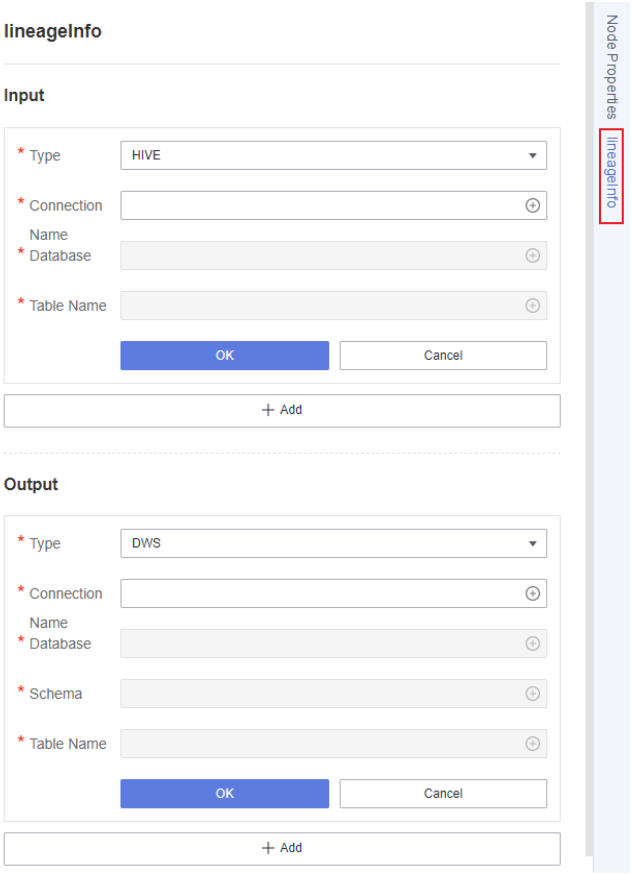
The following types of job nodes support manual lineage configuration.

- CDM Job

- **Rest Client**
- **DLI SQL**
- **DLI Spark**
- **DWS SQL**
- **MRS Spark SQL**
- **MRS Hive SQL**
- **MRS Presto SQL**
- **MRS Spark**
- **MRS Spark Python**
- **ETL Job**
- **OBS Manager**

When manually configuring the lineage, configure the input and output tables of the lineage on the Lineage tab page of the node. The data sources of the input and output tables can be DLI, DWS, Hive, CSS, OBS and CUSTOM. CUSTOM indicates a custom type. When manually configuring a lineage, you can add data sources that are not supported as custom types.

Figure 8-31 Example of manual configuration of lineage relationships

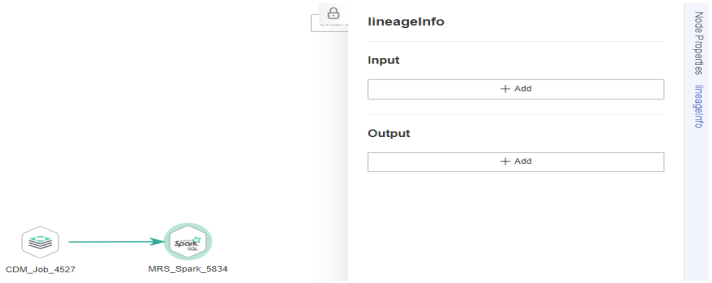


For example, you need to manually configure a lineage for an MRS Spark node in a pipeline data development job because this node does not support automatic lineage parsing. The procedure is as follows:

Step 1 On the DataArts Factory console, choose **Data Development** > **Develop Job**. Double-click the name of the job for which you want to configure a lineage to open the job canvas.

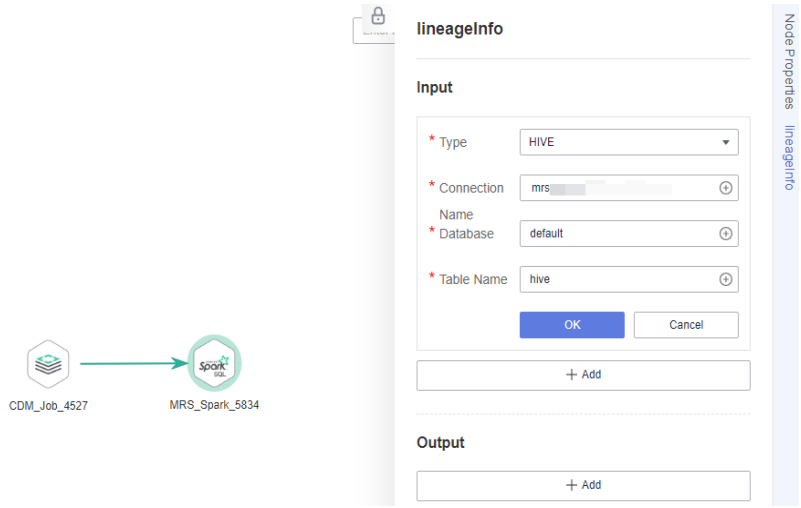
Step 2 Click the MRS Spark node in the job canvas and then the **lineageInfo** page.

Figure 8-32 lineageInfo page



Step 3 Configure the lineage input table. For example, you can configure input table **hive**, as shown in **Figure 8-33**.

Figure 8-33 Configuring the lineage input



Step 4 Click **OK** and configure the lineage output table. For example, you can configure output table **a**, as shown in **Figure 8-34**.

Figure 8-34 Configuring the lineage output

Step 5 Click **OK**. The lineage for the MRS Spark node has been configured. If you want to view the lineage later, collect metadata by referring to **Viewing Data Lineages** and schedule the job. Then, you can view the manually configured lineage of the MRS Spark node in DataArts Catalog.

----End

8.5.2.3 Viewing Data Lineages

You need to create a metadata collection task in DataArts Catalog first. When a data development job meets the **automatic lineage parsing requirements** or **lineages have been manually configured**, and when the job is successfully scheduled, you can view the data lineages in DataArts Catalog.

Constraints

- Data lineage updates depend on job scheduling. Data lineages are generated based on the latest job instances.
- To delete data lineages, you need to delete jobs or job metadata. Stopping jobs alone does not delete data lineages.

Creating and Running a Metadata Collection Task

Create and run a metadata collection task by referring to **Task Management**. When creating the task, select the tables whose lineages you want to view.

If a task for collecting the metadata of these tables has been created and run, skip this part.

Starting Job Scheduling

After metadata is collected, the system generates data lineages based on the latest job instances.

Step 1 On the DataArts Studio console, locate a workspace and click **DataArts Factory**.

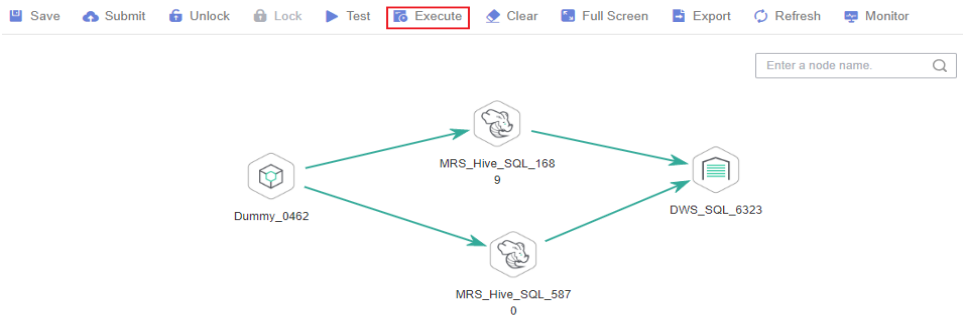
Step 2 In the navigation pane, click  and double-click the job for which lineages have been configured to open it.

Step 3 Click **Execute**. The system starts parsing lineages of the job.

NOTE

If you click **Test**, the system will not parse lineages of the job.

Figure 8-35 Starting job scheduling



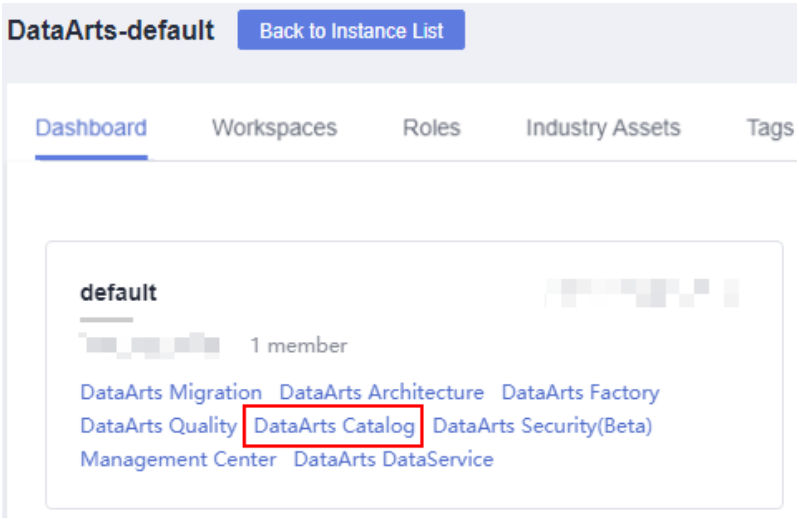
Step 4 After the job is successfully executed, wait for about 1 minute. The data lineage is generated.

----End

Viewing Data Lineages

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

Figure 8-36 DataArts Catalog



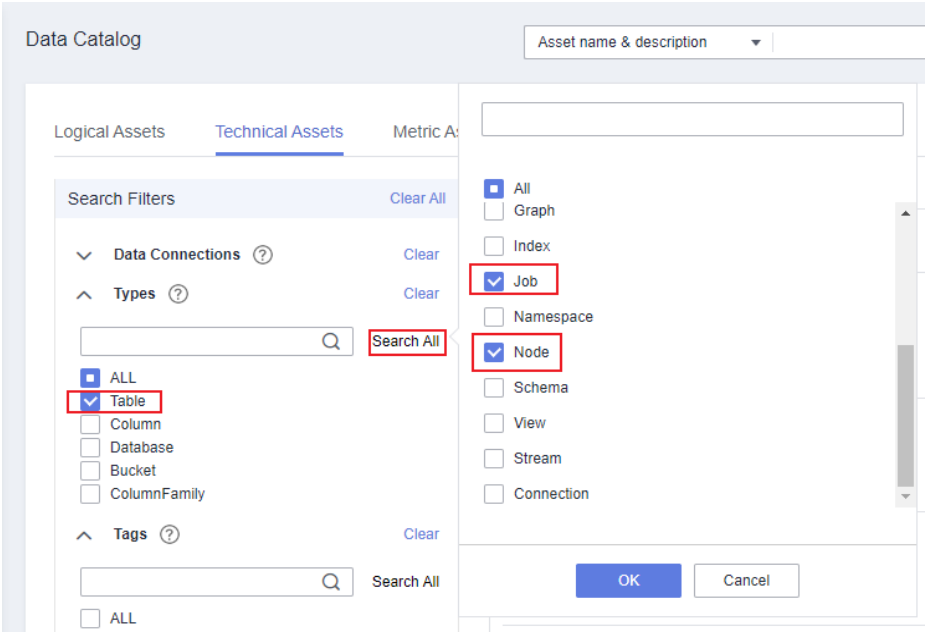
Step 2 In the navigation pane, choose **Data Catalog**. In the right pane, click the **Technical Assets** tab. On this page, you can query jobs, nodes, and tables.

In the **Types** area, click **Search All**, select **Job**, **Node**, and **Table**, and click **OK**.

NOTE

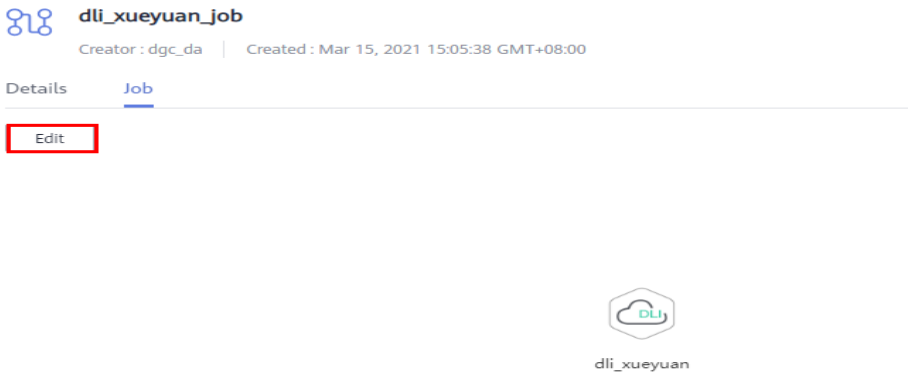
Jobs do not belong to any data connection. If you select a data connection in the search filters, no result will be returned.

Figure 8-37 Selecting types



Step 3 In the search result, click the name of an asset ending with **_job** to view its details. On the job details page, click the **Job** tab and then **Edit** to go to the job editing page.

Figure 8-38 Viewing job details

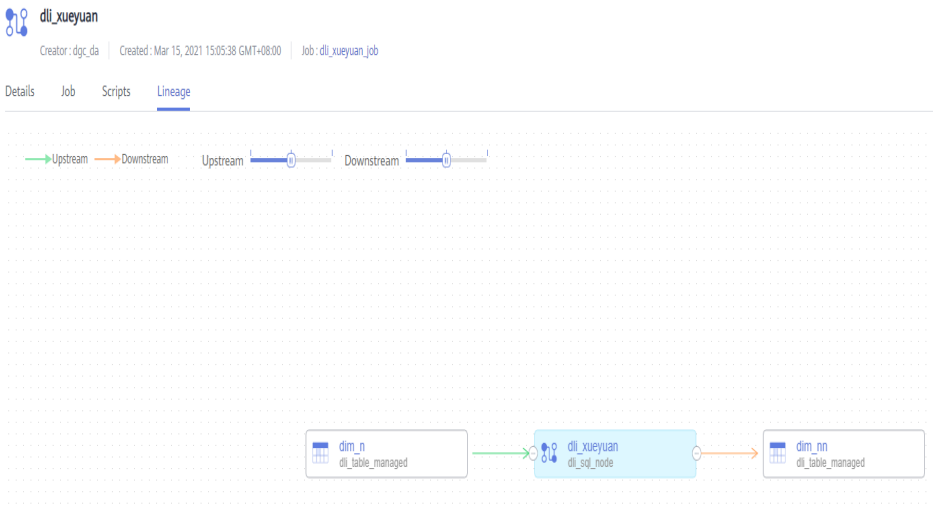


Step 4 In the data asset search result, click the name of an asset ending with **_node** to view its details. On the node details page, you can view the node lineage information.

- Click the **+** or **-** icon beside the node to expand its upstream and downstream links.

- Click a node to view the its details.
- Click the **Job** tab and then **Edit** to go to the job editing page.

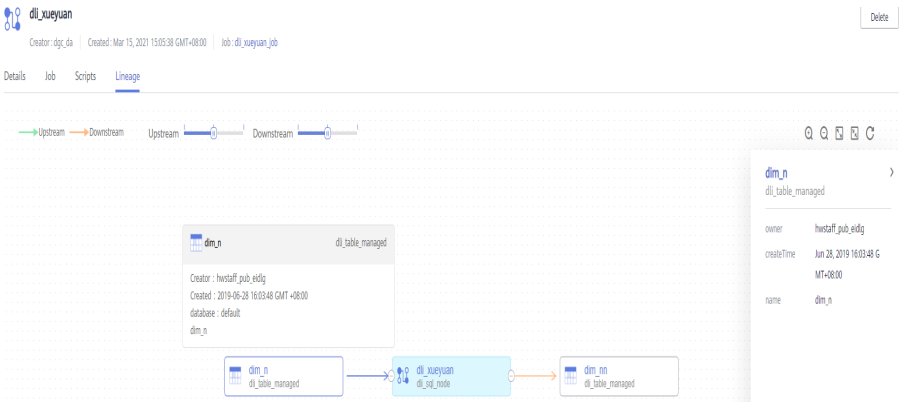
Figure 8-39 Viewing lineages of a node



Step 5 In the data asset search result, click the name of an asset whose icon is a table to view its details. On the table details page, you can view lineages of the table.

- Click the **+** or **-** icon beside the table to expand its upstream and downstream links.
- Click a table to view the its details.

Figure 8-40 Viewing lineages of a table



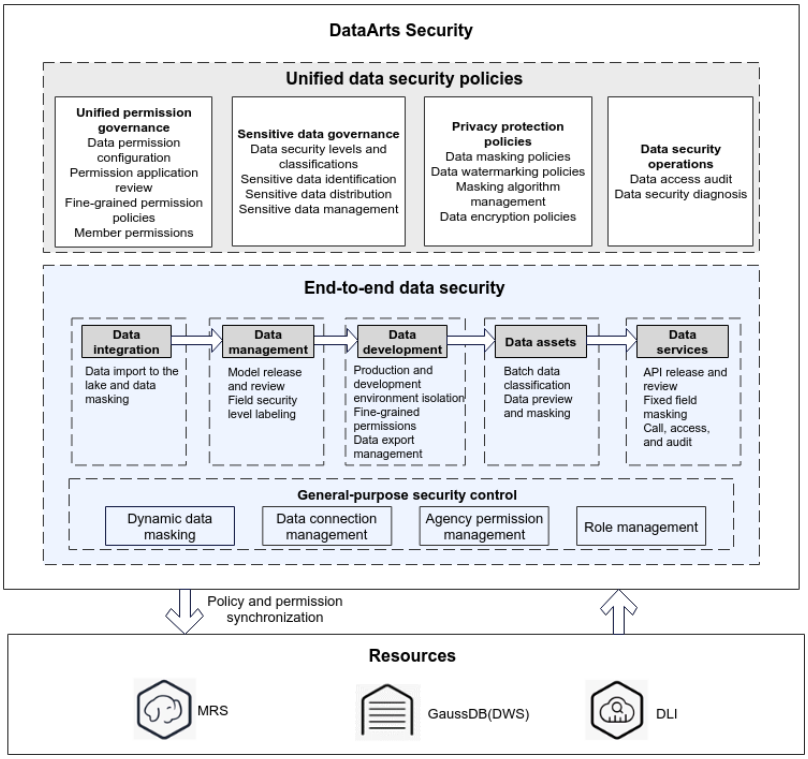
----End

9 DataArts Security

9.1 Overview

DataArts Security protects data lake security and meets the data security and governance requirements of different roles, such as data development engineers, data security administrators, data security auditors, and data security operators.

Figure 9-1 DataArts Studio DataArts Security framework



- **Resources**: databases, tables, fields, and computing engine queues in the Huawei Cloud data lake. They include the databases, tables, and fields of MRS Hive/Spark, DLI, and GaussDB(DWS), as well as computing queues of MRS Yarn and DLI.

- **End-to-end data security**: DataArts Studio protects data security throughout data integration, data management (architecture design, metric design, and data quality management), data development, data asset management, and data services. It protects data throughout its lifecycle and ensures secure data flow through measures such as data access control and data masking. For example, it can mask sensitive fields in the data to be imported to the data lake and can control access to data sources. When analysts query data, sensitive data can be protected using dynamic masking policies or field access permissions.
- **Unified data security policies**: unified permission governance, sensitive data governance, privacy protection policies, and data security operations.

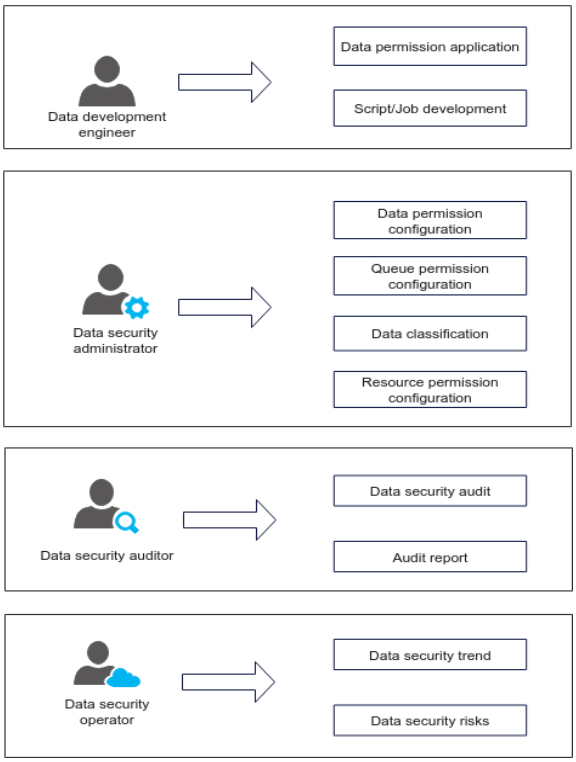
NOTE

DataArts Security is available in AP-Singapore, AP-Bangkok, AP-Jakarta, and AF-Johannesburg.

Scenario

DataArts Security meets the data security and governance requirements of different roles, such as data development engineers, data security administrators, data security auditors, and data security operators. **Figure 9-2** shows how different roles can use DataArts Security.

Figure 9-2 How different roles can use DataArts Security



Advantages

- DataArts Security integrates and centrally manages different big data services, such as MRS, DLI, and GaussDB(DWS), and provides unified permission configuration to improve usability and maintainability.

- DataArts Security provides end-to-end data security capabilities, such as unified permission governance, sensitive data governance, and privacy protection policy management.
- Unified permission governance allows you to allocate workspace permission sets (databases and tables that can be managed by each project workspace). You can assign permissions to users and user groups of different roles in a workspace. Cross-workspace dependency supports on-demand permission application, review, and approval.
- Sensitive data management supports classification, automatic discovery, and security management policies based on security levels of sensitive data.
- Privacy protection and management provides static and dynamic data masking and data watermarking capabilities to meet service requirements while ensuring data security.

Function

DataArts Security provides the following functions:

- **Unified permission governance**

  DataArts Security provides unified management of data permissions based on MRS, DLI, and GaussDB(DWS). You can create workspace permission sets, permission sets, or roles, and use them to control access to MRS, DLI, and GaussDB(DWS) data, assign the minimum permissions to users and user groups on demand, and reduce data security risks.
- **Sensitive data governance**

  You can create sensitive data identification rules (or rule groups), or use the built-in identification rules (or rule groups), to detect, classify, and grade sensitive data.
- **Privacy protection and management**

  You can use static and dynamic data masking, and data, file, and dynamic watermarking to prevent your data from being misused, disclosed, or stolen intentionally or unintentionally. In this way, your sensitive data is secure, complete, and safe to use.
- **Data security operations**

  DataArts Security provides data security diagnosis and data lake access and audit log query capabilities, helping you manage security better.

9.2 Dashboard

On the **Dashboard** page on the DataArts Security console, you can configure the data security administrator and view the number of sensitive tables, a pie chart of the security levels of sensitive tables, a pie chart of the security levels of sensitive fields, and the trends of the number of masking tasks and watermark embedding tasks.

Configuring the Security Administrator

The security administrator is specified by an account with the permissions of the DAYU Administrator system role. The security administrator has the highest

permissions in the DataArts Security module of all workspaces in the DataArts Studio instance. In the DataArts Security module, only the security administrator and DAYU Administrator have the permission to perform the following operations:

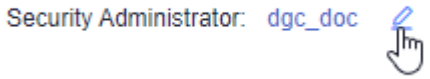
- Configuring workspace permission sets
- Configuring row-level access control using permissions
- Synchronizing users
- Configuring workspace resource permissions
- Configuring fine-grained authentication
- Configuring queue permissions

To configure the security administrator, log in to the DataArts Security console using an account with the permissions of the DAYU Administrator system role, and select an IAM user or user group on the **Dashboard** page. (If a user group is selected, all users in the user group are security administrators.)

NOTE

- Only the DAYU Administrator can configure a security administrator.
- The permissions of a security administrator take effect only for the DataArts Security component and are invalid for other components and services.

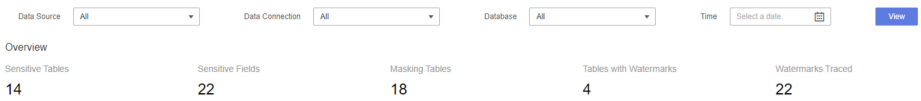
Figure 9-3 Configuring the security administrator



Viewing Sensitive Data

On the **Dashboard** page, you can filter data by data source and time. For example, you can view the sensitive data in the databases of GaussDB(DWS), DLI, and MRS Hive, including the number of sensitive tables, sensitive fields, masked tables, tables with watermarks, and watermarks traced.

Figure 9-4 Data overview



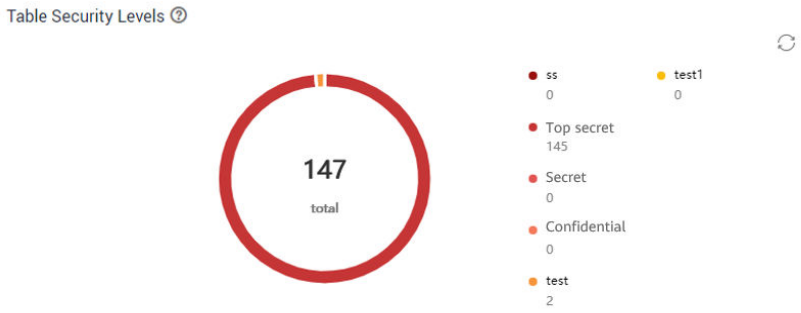
Data Analysis Reports

- Table security levels

  Create sensitive data discovery tasks to collect the number of table security levels. The security levels are customizable. The number of custom security levels and associated sensitive tables are displayed beside the pie chart.

  For details on how to create and run a sensitive data discovery task, see **Creating a Sensitive Data Discovery Task**.

Figure 9-5 Security level pie chart

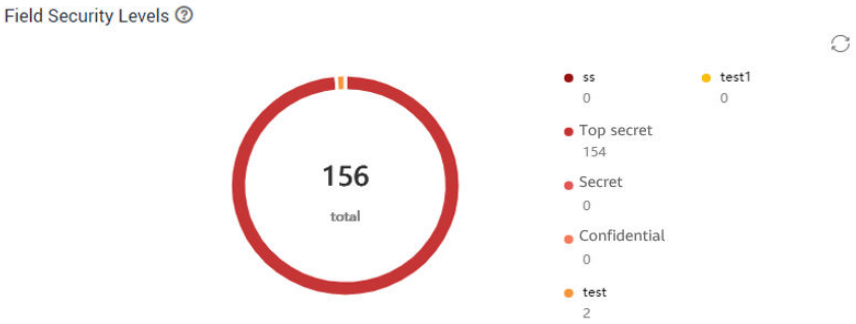


- Field security levels

  Create sensitive data discovery tasks to detect sensitive table fields. The field security levels are customizable. The number of custom security levels and associated sensitive fields are displayed beside the pie chart.

  For details on how to create and run a sensitive data discovery task, see **Creating a Sensitive Data Discovery Task**.

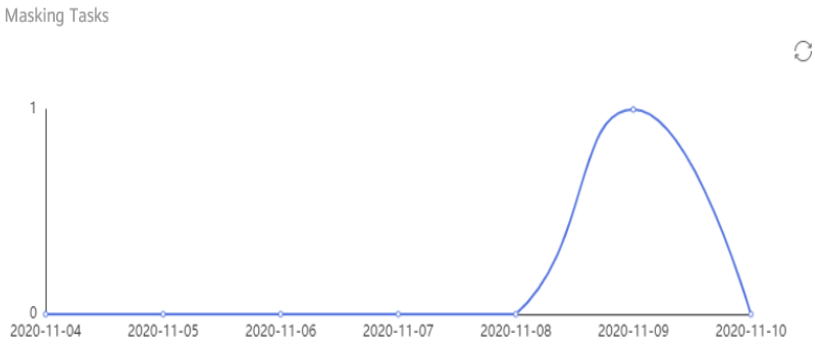
  Figure 9-6 Security level pie chart



- Masking tasks

  The number of masking tasks on each day in the last seven days is displayed. For details on how to create and run a data masking task, see **Create a Static Masking Task**.

Figure 9-7 Changes of masking task quantity

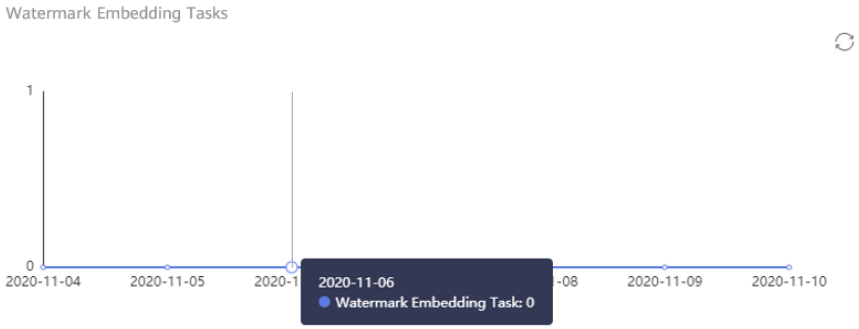


- Watermark embedding tasks

  The number of watermark embedding tasks on each day in the last seven days is displayed.

  For details on how to create and run a watermark embedding task, see **Creating a Data Watermark Embedding Task**.

  Figure 9-8 Changes of watermark embedding task quantity



9.3 Unified Permission Governance

9.3.1 Permission Governance Process

Unified permission governance allows you to configure access permissions for the databases, tables, and fields in MRS, DLI, and GaussDB(DWS). It has the following features:

- Centralized access control

  Permissions of different big data services, such as MRS, DLI, and GaussDB(DWS), are centrally managed. A unified portal is available for you to configure and maintain permissions easily.

- Multi-level permission configuration model

Permission models are clearly defined and managed by level. A permission set or role further splits the permission scope defined by the workspace permission set and associates users with permissions for permission control.

- Refined permission management

  Role-based access control (RBAC) on the console supports refined data permission configuration and permission assignment by role, user, and user group. In addition, on-demand and efficient permission application approval is supported. Approved permissions take effect immediately.

- Multi-dimensional permission display
  - By workspace member: You can display the data table permissions requested by each user or user group and display, configure, and revoke the permission set relationship of each user or user group.
  - By data: You can display and configure the permission relationships of data in the current permission set by database, table, or field.
  - By permission: You can display, configure, and revoke the permission policy relationships of data in the current permission set by permission policy.

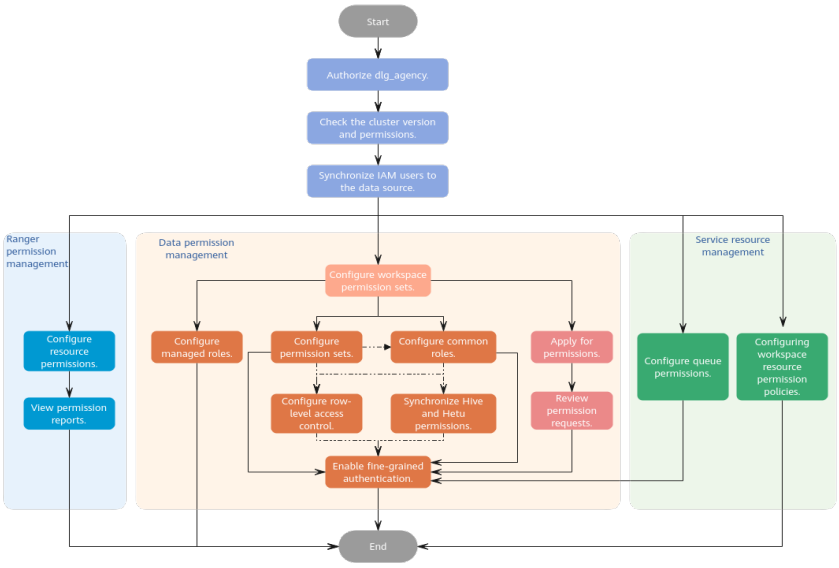
- Workspace resource management

  In addition to data permissions, workspace resources, such as data connections and agencies, can be managed.

Use Process

Figure 9-9 shows the process for using unified permission governance.

Figure 9-9 Process of using unified permission governance



Unified permission governance supports **data permission management**, **service resource management**, and **Ranger permission management**. Their processes are as follows:

Data permission management process

1. **Authorize dlg_agency.**

   When using an agency, DataArts Security requires higher cloud service permissions. Before using DataArts Security, you need to grant required permissions to dlg_agency.

2. **Check the cluster version and permissions.**

   Unified permission governance has requirements on the data connection agent, data source version, and user permissions. Before using it, you need to check and prepare related configurations.

3. **Synchronize IAM users to the data source.**

   Synchronize user information from IAM to data sources so that users' access to the data sources can be managed based on user information.

4. **Configure workspace permission sets.**

   As the largest permission set in a DataArts Studio workspace, the workspace permission set defines the resources that can be accessed by users in the workspace.

5. **Configure permission sets.**

   A permission set associates users with permissions. You can create multiple permission sets to associate users in different scenarios with different permissions. Permissions can be managed through permission synchronization (association of permission sets with roles are more recommended in actual applications.)

6. **Configure common roles.**

   Create roles in the data source to associate users and permissions. In this way, you can manage permissions more intuitively.

7. **Configure managed roles.**

   Manage the existing roles in the MRS data source and inherit the MRS data source permissions of the existing roles.

8. **Configure row-level access control.**

   DataArts Security allows you to view the permissions of workspace members, and manage roles and permission sets.

9. **Synchronize MRS Hive and Hetu permissions.**

   DataArts Security allows you to view the permissions of workspace members, and manage roles and permission sets.

10. **Apply for permissions.**

    During access permission management, you can grant permissions to users through permission sets or roles, or apply for permissions and approve permission applications.

11. **Review permission requests.**

    The approver is the administrator of the permission set or role. The requested permission takes effect immediately after being approved.

12. **Enable fine-grained authentication.**

    After fine-grained authentication is enabled, data sources no longer use the accounts of the data connections during script execution, job tests, and job scheduling in DataArts Factory of DataArts Studio. Instead, the current user is used for authentication. In this way, different users have different data permissions, and the permissions of roles, permission sets, and queues can be managed.

Service resource management process

1. **Configure queue permissions.**

   Queue permissions can be used to allocate MRS Yarn and DLI queues to the current workspace and configure queue permission policies for user groups or users.

   - After queues are allocated to the workspace, they can be selected during the job node configuration in DataArts Factory.
   - After queue permission policies are configured for user groups or users, they have the permissions specified in the policies.

2. **Configure workspace resource permission policies.**

   DataArts Security supports management of workspace resources, such as data connections and agencies. Unauthorized users cannot view or use the resources.

Ranger permission management process

1. **Configure resource permissions.**

   You can create permission policies for MRS components and use the Ranger component to manage permissions.

2. **View permission reports.**

   You can view resource permission policies and their details through a comprehensive permission report.

Data Permission Management

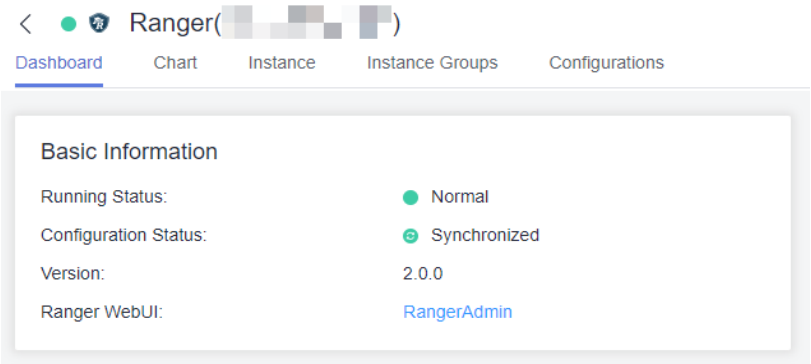
The current data permission control uses the allowlist mechanism, which adds operation conditions to the users to be authorized without affecting the permissions the users already have. If you only want to make the permissions granted by the data permission control take effect, you need to revoke the original permissions of the users to be authorized. By default, DataArts Studio users have the following permissions:

- For DLI data sources, the DAYU Administrator or DAYU User has the DLI Service Admin permission by default. Therefore, the users to be authorized have all the data permissions of DLI database tables by default. To remove the default permissions of an authorized user, you need to delete the DLI Service Admin permission of the user.
- For GaussDB(DWS) data sources, even if the DAYU Administrator or DAYU User has the GaussDB(DWS) Administrator permission by default, the GaussDB(DWS) database permissions are isolated from the IAM permissions on the console. Therefore, the users to be authorized do not have the data permissions of GaussDB(DWS) database tables by default. Only the data permission granted by the current data permission control takes effect.
- For MRS data sources, the DAYU Administrator or DAYU User has the MRS Administrator permission by default. After synchronized to MRS, the user is assigned a role. For details, see **Synchronizing IAM Users to MRS**. Then the Ranger component enables permissions for the default policy. For details, see **Configuring Component Permission Policies**. Therefore, the users to be authorized have the data permissions of MRS Hive database tables by default. If you want to revoke the default permissions of the users to be authorized, perform the following operations to remove the **public** user group from the default system policies on the Ranger component:

a. Log in to MRS Manager as user **admin**.

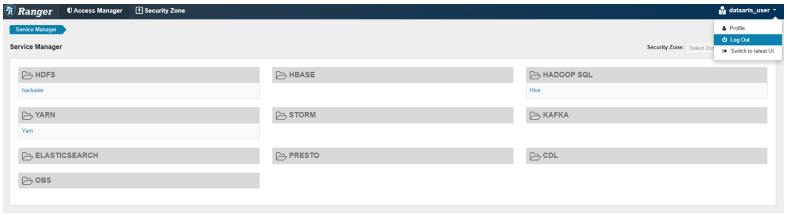
b. On the Manager page, choose **Cluster** > **Services** > **Ranger**. On the Ranger overview page, click **RangerAdmin** to go to the Ranger WebUI.

Figure 9-10 Accessing the Ranger WebUI



c. Log out of the current account and use the Ranger administrator account to log in again. For a common cluster, the admin account for the Manager page can be used as the Ranger administrator account. For a security cluster, **rangeradmin** is the Ranger administrator account. For details about the default password of **rangeradmin**, see **User Account List**.

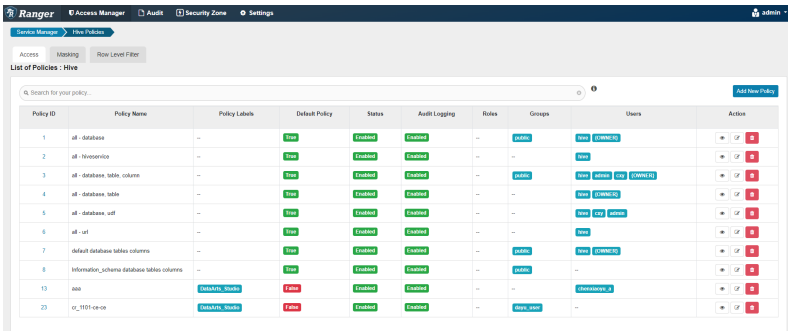
Figure 9-11 Logging out of the current account



d. On the home page, click the component plug-in name in the **HADOOP SQL** area, for example, **Hive**.

e. On the **Access** tab page, locate the default policies whose **Groups** column contains **public** (that is, the policy whose value in the **Default Policy** column is **True**) and remove the **public** user group from the policies.

Figure 9-12 Policy list



9.3.2 Authorizing dlg_agency

Cloud service agencies allow DataArts Studio to perform operations such as task scheduling and resource O&M on cloud services on your behalf. When you log in to the DataArts Studio console homepage for the first time, a dialog box is displayed, prompting you to authorize other cloud services to access DataArts Studio. After the authorization is complete, DataArts Studio automatically creates an agency named **dlg_agency**. If you do not agree to the authorization, the dialog box will be displayed again when you access the console homepage next time.

When using an agency, DataArts Security requires higher cloud service permissions. Before using DataArts Security, you need to grant the permissions listed in **Table 9-1** to **dlg_agency**.

Table 9-1 Required permissions

Permission	Purpose	Mandatory	Authorization Item/System Permission (Configure Either of Them)	
IAM permission	This permission is required for the system to obtain users or user groups, or create roles. For example, user synchronization fails if this permission is missing.	Mandatory for MRS, Gauss DB(DWS), and DLI permission management	• iam:users:listUsers • iam:groups:listGroups • iam:users:listUsersForGroup • iam:roles:createRole • iam:roles:deleteRole • iam:roles:updateRole • iam:permissions:grantRoleToGroup • iam:permissions:listRoleAssignments • iam:permissions:revokeRoleFromGroup	Security Administrator

Permission	Purpose	Mandatory	Authorization Item/System Permission (Configure Either of Them)	
MRS/ GaussDB(DWS) data connection agent permission	This permission is required for permission synchronization. For example, if this permission is missing, permission synchronization between permission sets, role permission synchronization, or permission application approval fails.	Mandatory for MRS and GaussDB(DWS) permission management	Any CDM permission, for example, cdm:cluster:get	Any CDM permission, for example, CDM Administrator
MRS user synchronization permission	This permission is required for MRS user synchronization. For example, MRS user synchronization fails if this permission is missing.	Mandatory for MRS permission management	• mrs:cluster:syncUser	MRS FullAccess
GaussDB(DWS) user synchronization permission	This permission is required for GaussDB(DWS) user synchronization. For example, GaussDB(DWS) user synchronization fails if this permission is missing.	Mandatory for GaussDB(DWS) permission management	• dws:dbAuthority:syncIamUser • dws:dbAuthority:updateUser	DWS FullAccess

Permission	Purpose	Mandatory	Authorization Item/System Permission (Configure Either of Them)	
DLI permission synchronization permission	This permission is required for DLI permission synchronization. For example, if this permission is missing, DLI permission synchronization fails and the system displays a message indicating insufficient permissions.	Mandatory for DLI permission management	• dli:database:grantPrivilege • dli:table:grantPrivilege • dli:column:grantPrivilege • dli:queue:grantPrivilege	DLI FullAccess

Prerequisites

In the dialog box displayed on the DataArts Studio console homepage, you have selected **Agree** to allow the system to automatically create an agency named **dlg_agency**.

Constraints

After the agency authorization is successful, it takes 15 to 30 minutes for the permissions to take effect. Then, you can use DataArts Security to manage access permissions.

Granting Permissions to dlg_agency

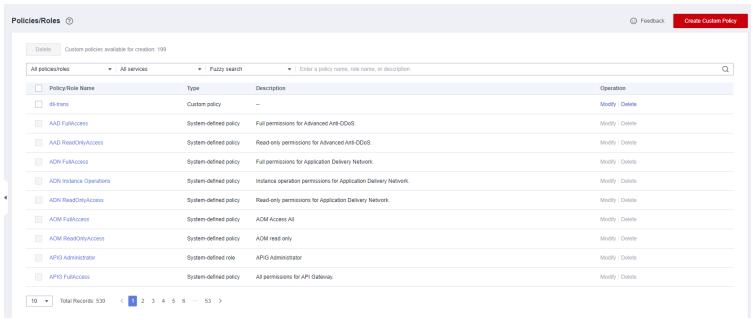
When granting permissions to **dlg_agency**, you need to select either an authorization item or system permission from **Table 9-1** as needed.

This section uses the MRS permission management scenario as an example. The permissions to be granted include the IAM permission, MRS/GaussDB(DWS) data connection agent permission, and MRS user synchronization permission. The principle of least privilege is used. The operations are as follows:

Step 1 Log in to the IAM console.

Step 2 In the left navigation pane, choose **Permissions** > **Policies/Roles** and click **Create Custom Policy**.

Figure 9-13 Clicking Create Custom Policy



Step 3 On the displayed **Create Custom Policy** page, select **JSON** for **Policy View**, enter the IAM custom policy content required for MRS permission management, and click **OK**.

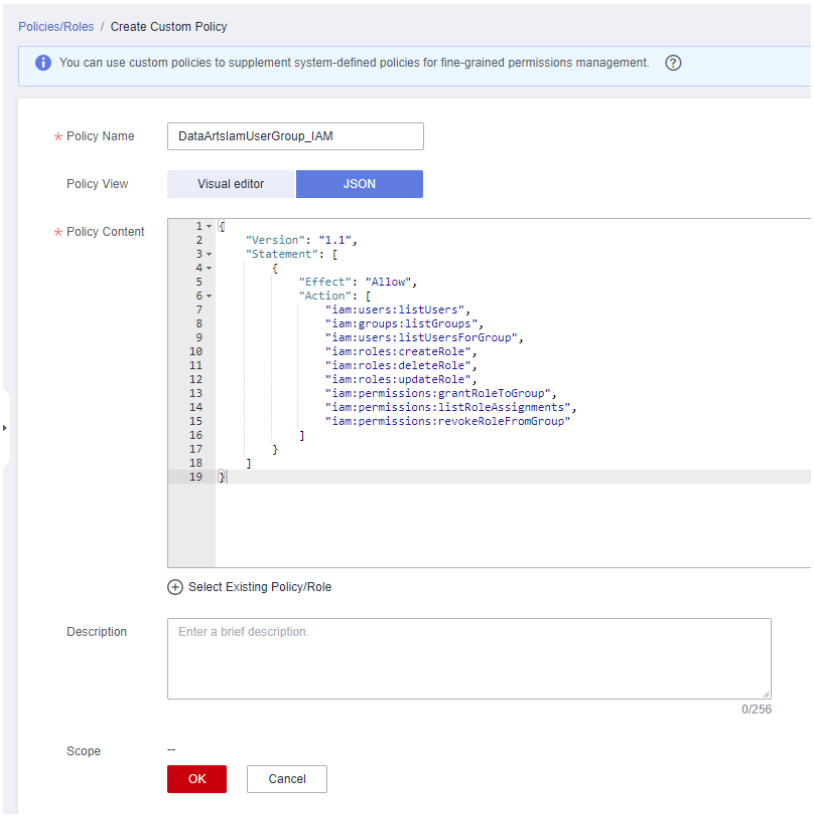
NOTE

A custom policy can contain permissions for either global or project-level services. You need to configure IAM policies first, and then MRS and CDM policies.

- **Policy Name**: Enter **DataArtsIamUserGroup_IAM**.
- **Policy View**: Select **JSON** to switch to the JSON view.
- **Policy Content**: Enter the following JSON code and click **OK**.

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "iam:users:listUsers",
            "iam:groups:listGroups",
            "iam:users:listUsersForGroup",
            "iam:roles:createRole",
            "iam:roles:deleteRole",
            "iam:roles:updateRole",
            "iam:permissions:grantRoleToGroup",
            "iam:permissions:listRoleAssignments",
            "iam:permissions:revokeRoleFromGroup"
         ]
      }
   ]
}
```


Figure 9-14 Creating a custom policy for IAM



Step 4 Click **Create Custom Policy** again. On the displayed **Create Custom Policy** page, select **JSON** for **Policy View**, enter the MRS and CDM custom policy content required for MRS permission management, and click **OK**.

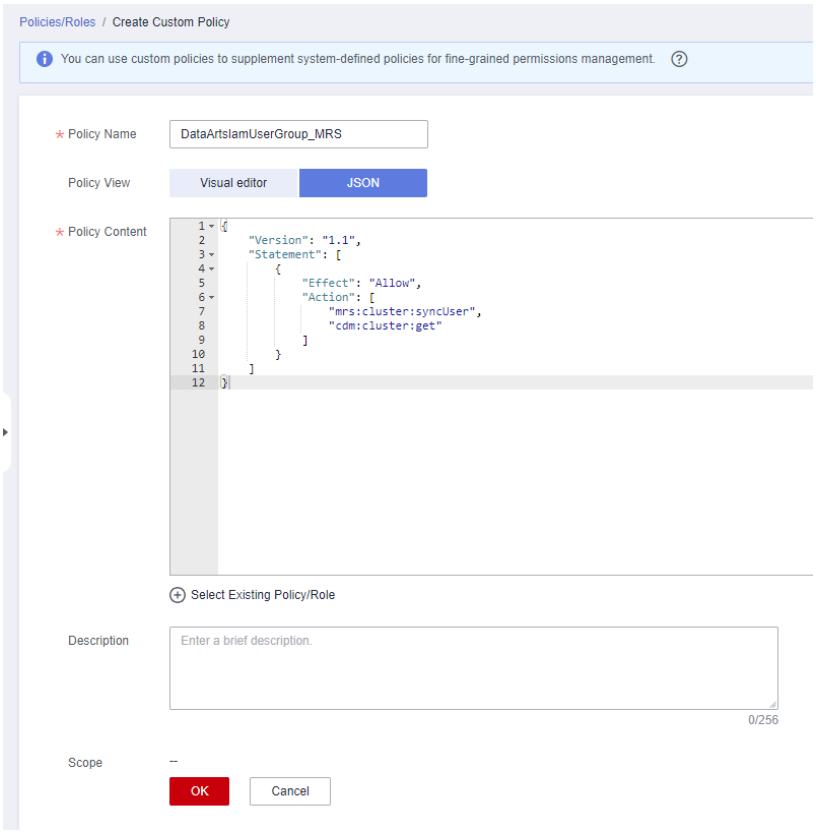
NOTE

A custom policy can contain permissions for either global or project-level services. You need to configure IAM policies first, and then MRS and CDM policies.

- **Policy Name**: Enter **DataArtsIamUserGroup_MRS**.
- **Policy View**: Select **JSON** to switch to the JSON view.
- **Policy Content**: Enter the following JSON code.

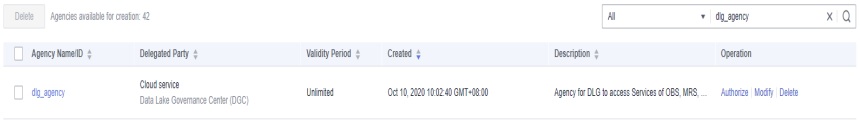
```
{
  "Version": "1.1",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "mrs:cluster:syncUser",
        "cdm:cluster:get"
      ]
    }
  ]
}
```


Figure 9-15 Creating custom policies for MRS and CDM



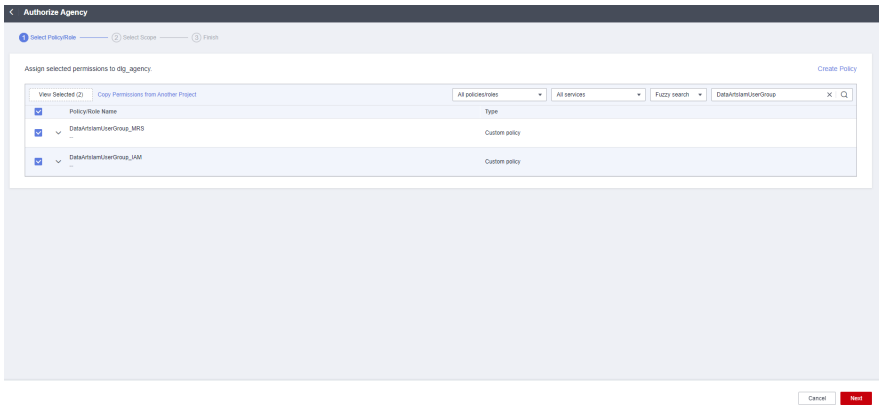
Step 5 In the left navigation pane, choose **Agencies**, search for **dlg_agency**, and click **Authorize**.

Figure 9-16 Authorizing dlg_agency



Step 6 In the displayed dialog box, locate and select the created custom policies **DataArtsIamUserGroup_IAM** and **DataArtsIamUserGroup_MRS**, and click **Next**.

Figure 9-17 Selecting the created custom policies



Step 7 Click **OK**. After the authorization is complete, wait for 15 to 30 minutes. Then, you can use DataArts Security to manage MRS access permissions.

----End

9.3.3 Checking the Cluster Version and Permissions

Unified permission governance has requirements on the data connection agent, data source version, and user permissions. Before using it, you need to check and prepare related configurations based on **Table 9-2**.

NOTE

DLI permission management involves only **Authorizing dlg_agency** and does not involve cluster version and permissions check.

Checklist

Table 9-2 Checklist

Check Item	Mandatory	Check Content	Configuration Guide
Data connection agent version	Mandatory for MRS/ GaussDB(DWS) permission management	The CDM cluster version is 2.10.0.300 or later.	Log in to the CDM console and click **Cluster Management**. In the cluster list, locate the required cluster and click the cluster name. On the **Basic Information** page, view the cluster version. If the version is not the required one, create another CDM cluster of the latest version or contact customer service or technical support.

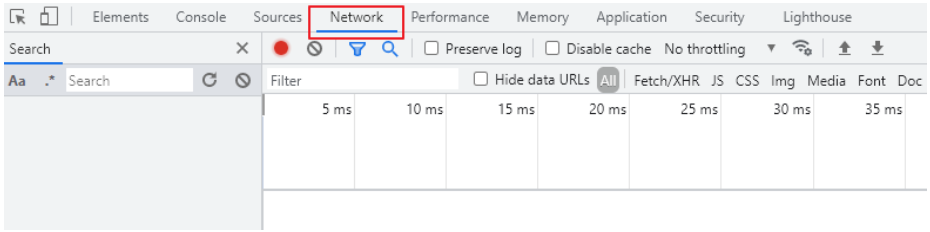
Check Item	Mandatory	Check Content	Configuration Guide
Ranger component configuration	Mandatory for MRS permission management	LDAP user synchronization is enabled for the Ranger component of an MRS non-security cluster.	In a non-security MRS cluster, the Ranger component synchronizes Unix users by default, but does not synchronize users, user groups, or roles on Manager. Therefore, you need to switch the user synchronization policy. For details, see **Configuring the Ranger Component**.
Ranger connection user permission		The user for the connection has the admin permission of the Ranger component.	The user for the Ranger connection must have the admin permission of the Ranger component. For details, see **Preparing a Ranger Admin User**.
guest_agent version of the GaussDB(DWS) cluster	Mandatory for GaussDB(DWS) permission management	The guest_agent version of the GaussDB(DWS) cluster is 8.2.1, or later than 8.2.1 and earlier than 9.0.0.	You can view the guest_agent version of the GaussDB(DWS) cluster using a developer debugging tool. For details, see **Viewing the guest_agent Version of a GaussDB(DWS) Cluster**.
GaussDB(DWS) connection user permissions		• In the non-RSM mode, the user for the connection must have at least the dbadmin permission of the database. • In the RSM mode, the user must have the system administrator permissions.	• In the non-RSM mode, set the dbadmin administrator by referring to **Database Users**. • In the RSM mode, set the system administrator by referring to **Configuring Separation of Permissions**.

Viewing the guest_agent Version of a GaussDB(DWS) Cluster

Step 1 Log in to the GaussDB (DWS) console, choose **Clusters**, and locate a cluster.

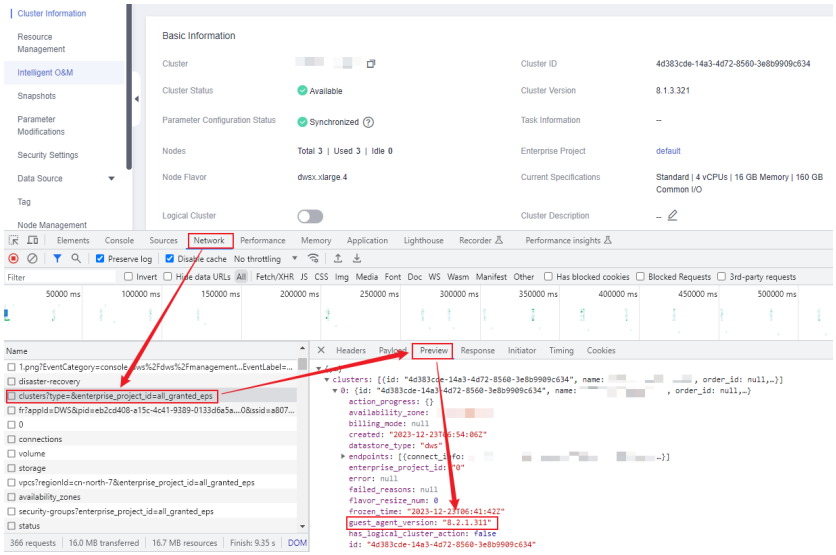
Step 2 Press **F12** to open the developer debugging tool and click the **Network** tab.

Figure 9-18 Network



Step 3 Click the name of the cluster to go to the **Basic Information** page. On the **Network** tab page, locate and click the long string starting with **clusters?type=xxxxxx**. In the right pane, click **Preview** and search for the **guest_agent_version** field, whose value is the guest_agent version of the GaussDB(DWS) cluster.

Figure 9-19 Locating the guest_agent_version field



Step 4 If the version is not your required one, contact the customer service or technical support of GaussDB(DWS).

----End

Configuring the Ranger Component

In a non-security MRS cluster, the Ranger component synchronizes Unix users by default, but does not synchronize users, user groups, or roles on FusionInsight Manager. Therefore, you need to switch the user synchronization policy. The procedure is as follows:

NOTE

By default, the Ranger component of an MRS security cluster synchronizes LDAP users. No additional operation is required. If the default configuration is changed, you can change the user synchronization policy by referring to this section.

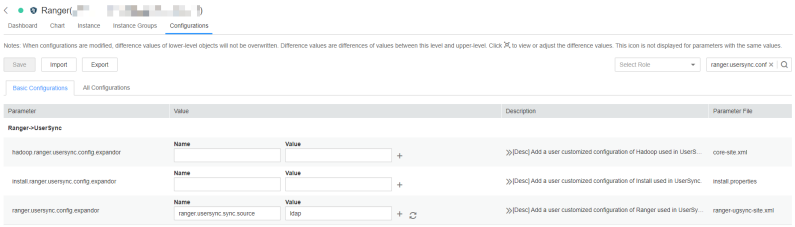
Step 1 Log in to MRS Manager as user **admin**.

Step 2 On the Manager page, choose **Cluster** > **Services** > **Ranger** > **Configurations** > **Basic Configurations**, search for **ranger.usersync.config.expandor** in the search box, and set its name to **ranger.usersync.sync.source** and value to **ldap**.

NOTE

By default, this parameter is unavailable for MRS clusters of old versions (for example, MRS 3.1.0). You can contact the customer service or technical support of MRS for support.

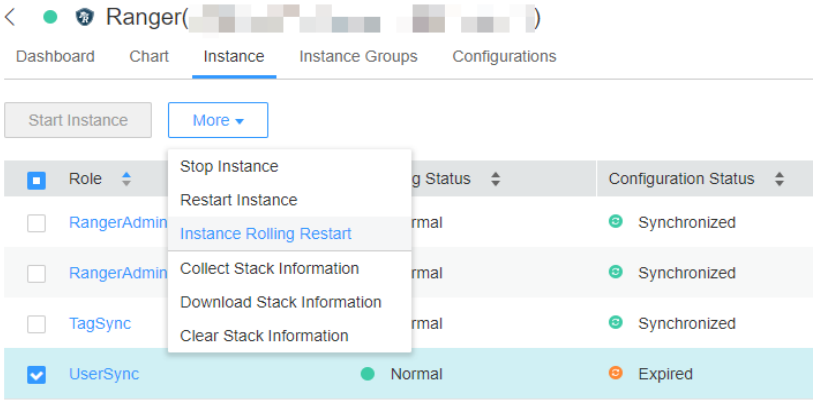
Figure 9-20 Configuring the ranger.usersync.config.expandor parameter



Step 3 After the parameter is set, click **Save** in the upper left corner and then **OK** in the dialog box to save the configuration.

Step 4 After the configuration is saved, switch to the **Instances** tab page, select the **UserSync** instance that has expired, click **More**, and select **Instance Rolling Restart** to make the configuration take effect.

Figure 9-21 Performing a rolling instance restart



----End

Preparing a Ranger Admin User

The user for the Ranger connection must have the admin permission of the Ranger component. The procedure is as follows:

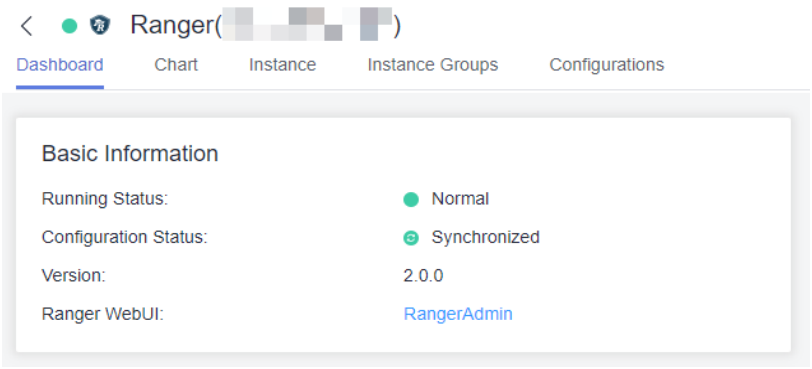
Step 1 Log in to MRS Manager as user **admin**.

Step 2 Choose **System** > **Permission** > **User**. On the page displayed, click **Create** to add a dedicated user as the Kerberos authentication user. Select user groups **superGroup** and **hive** for the user, and assign the **Manager_administrator** role to the user.

Step 3 Log in to MRS Manager as the new user and change the initial password.

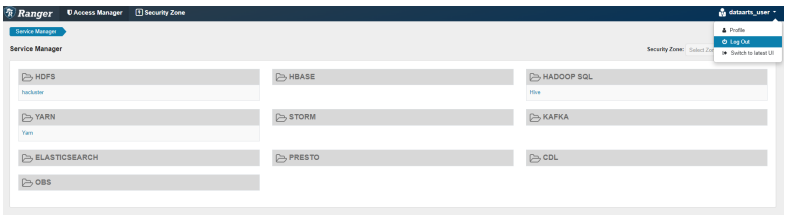
Step 4 On the Manager page, choose **Cluster** > **Services** > **Ranger**. On the Ranger overview page, click **RangerAdmin** to go to the Ranger WebUI.

Figure 9-22 Accessing the Ranger WebUI



Step 5 Log out of the current account and use the Ranger administrator account to log in again. For a common cluster, the admin account for the Manager page can be used as the Ranger administrator account. For a security cluster, **rangeradmin** is the Ranger administrator account. For details about the default password of **rangeradmin**, see **User Account List**.

Figure 9-23 Logging out of the current account

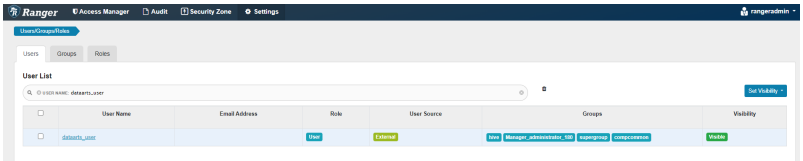


Step 6 Change the role of the new user from Ranger to Admin. Find the name of the new user in **Settings** > **Users/Groups/Roles** > **Users**.

NOTE

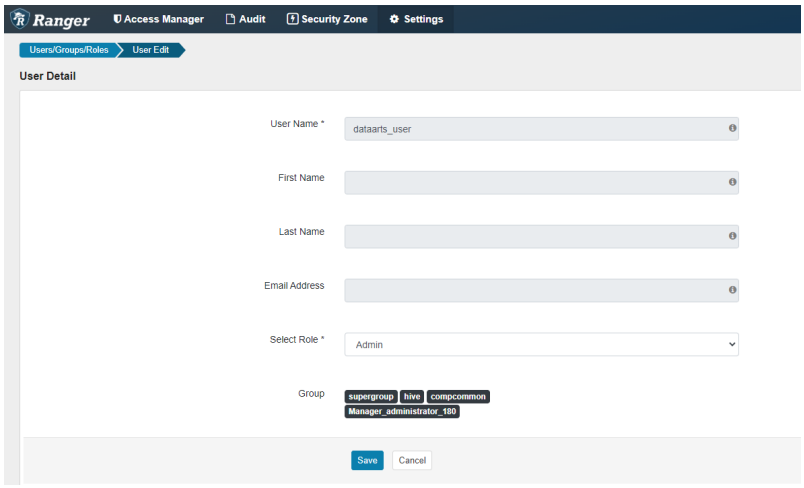
If the new user is not found on Ranger, wait for about five minutes until Ranger automatically synchronizes the MRS cluster role.

Figure 9-24 Searching for the username



Step 7 Click the username to go to the details page, change the user role to **Admin**, and click **Save**.

Figure 9-25 Changing the user role



----End

9.3.4 Synchronizing IAM Users to the Data Source

By default, when a user accesses the MRS or GaussDB(DWS) data source through a data connection in DataArts Studio, the username and password in the data connection are used for authentication. To manage users' data access permissions based on user information, you need to synchronize user information from IAM to the data source so that different users have different identities in the data source and can use their own user information for authentication during data permission management.

Note that each MRS/GaussDB(DWS) cluster in a DataArts Studio instance can have only one user synchronization task. User synchronization tasks are configured at the DataArts Studio instance level, and data can be exchanged between workspaces.

Prerequisites

- You have created a GaussDB(DWS) or MRS Ranger data connection in Management Center. For details, see **Creating a Data Connection**.
- You have configured permissions for the **dlg_agency** by referring to **Authorizing dlg_agency**.

Constraints

- In a DataArts Studio instance, each MRS/GaussDB(DWS) cluster can have only one user synchronization task.
- If a user synchronization task keeps running for more than half an hour, the task will be stopped due to timeout. If the synchronization fails for more than 10 consecutive times, the task will be stopped.
- Federated users have only user group information and cannot be synchronized.
- The data source synchronizes only the user information of its own tenant and cannot synchronize the clusters from data sources of other tenants who are not connected through IP connections.

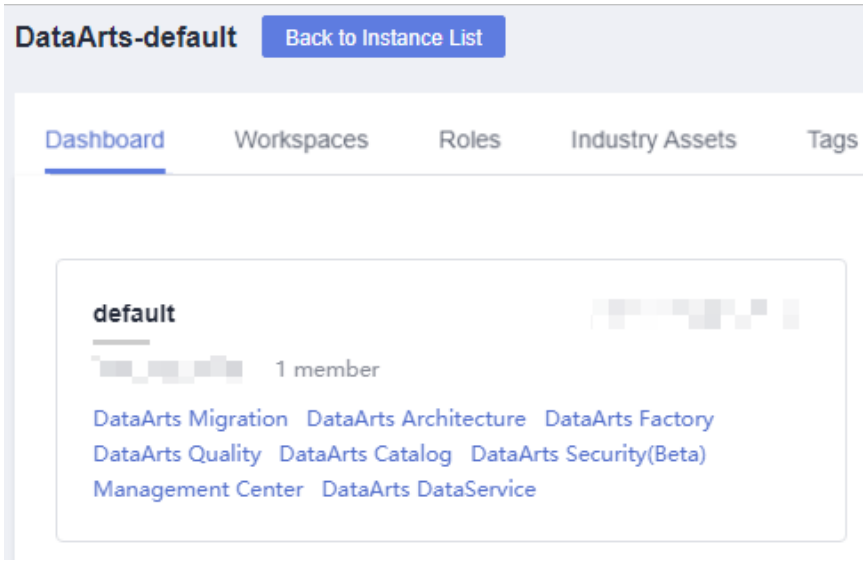
- User synchronization is available only for MRS Hive and GaussDB(DWS) data sources. For the GaussDB(DWS) data source, user synchronization is mandatory. For the MRS data source, you can create MRS users with the same name as IAM users so that user synchronization is not needed. DLI uses IAM users for authentication. Therefore, user synchronization is not required.
- The restrictions on MRS user synchronization tasks are as follows:
  - If a human-machine user with the same name as the user to be synchronized exists in MRS, the MRS user synchronization task fails. No error message is displayed for this error. You are advised to use either of the following methods to resolve this issue:
    - Use the IAM user synchronization function on the MRS cluster details page, rather than run the user synchronization task on the DataArts Security console again. The IAM user synchronization function is similar to the user synchronization task, except that if there are users with the same name, only these users will fail to be synchronized, and the other users can still be synchronized successfully.
    - Log in to MRS Manager, choose **System** > **Permission** > **User**, and delete the human-machine user with the same name as the user to be synchronized.
    - On the IAM console, delete the user to be synchronized with the same name as the MRS human-machine user.
  - Before MRS data source synchronization, ensure that the user or user group has at least one of the following permissions. Otherwise, the user or user group will not be synchronized.
    - Tenant Administrator
    - MRS FullAccess
    - MRS CommonOperations
    - MRS ReadOnlyAccess
    - MRS Administrator
    - MRS Admin
    - MRS User
    - MRS Viewer
    - Self Define (any custom policy)
- The restrictions on GaussDB(DWS) user synchronization tasks are as follows:
  - GaussDB(DWS) user synchronization is supported only when the guest_agent version of a GaussDB(DWS) cluster is 8.2.1, or later than 8.2.1 and earlier than 9.0.0. For details about how to check the guest_agent version of a GaussDB(DWS) cluster, see **Viewing the guest_agent Version of a GaussDB(DWS) Cluster**.
  - Before GaussDB(DWS) user synchronization, ensure that the users at least have the DWS Database Access permission. Otherwise, the synchronization will fail.

- To synchronize IAM users to GaussDB(DWS), you must configure the following permissions for the **dlg_agency**. For details, see **Authorizing dlg_agency**.
  - dws:dbAuthority:syncIamUse
  - iam:users:listUsers
  - iam:groups:listGroups
  - iam:users:listUsersForGroup
- GaussDB(DWS) does not support user groups. When an IAM user group is synchronized to GaussDB(DWS), a user named in the *iam_group_**User group ID*** format will be created in GaussDB(DWS), and the *iam_group_**User group ID*** user corresponding to the user group deleted from IAM will be deleted from GaussDB(DWS). Do not create a user prefixed with **iam_group_** on GaussDB(DWS) because such a user may be deleted by mistake.

Creating a User Synchronization Task

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

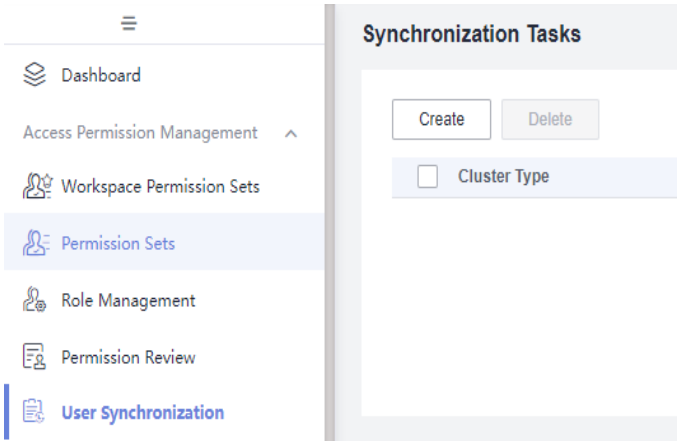
Figure 9-26 DataArts Security



Step 2 In the left navigation pane on the DataArts Security console, choose **User Synchronization**.

Step 3 On the **Synchronization Tasks** page, click **Create** to create a user synchronization task.

Figure 9-27 Creating a user synchronization task



Step 4 For details about how to set parameters for creating a user synchronization task, see **Table 9-3**. After setting the parameters, click **OK** to create a user synchronization task.

Figure 9-28 Creating a user synchronization task

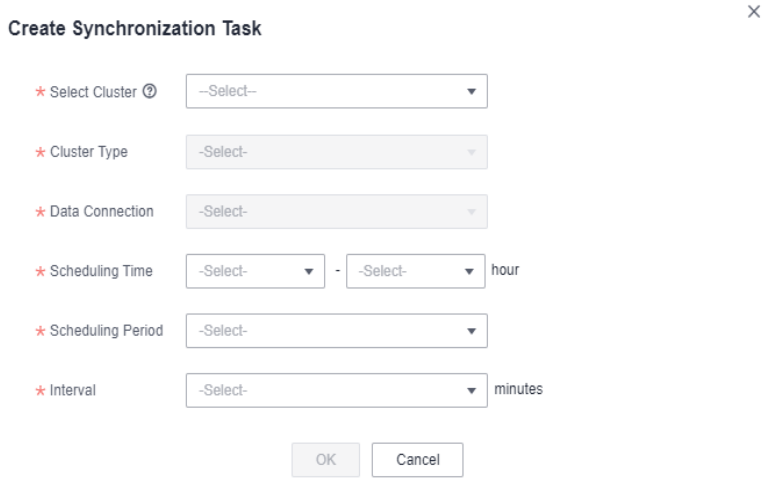


Table 9-3 Parameters for creating a user synchronization task

Parameter	Description
*Select Cluster	Select a GaussDB(DWS) or MRS cluster that is connected through a GaussDB(DWS) or MRS data connection.
*Cluster Type	You do not need to set this parameter. A matched cluster type is automatically selected.
*Data Connection	You do not need to set this parameter. The data source cluster is automatically selected.

Parameter	Description
*Scheduling Time	Select the time period for scheduling, with the start time included and end time excluded. For example, if the scheduling time is set to **00-05**, the task runs from 00:00 to 05:00 every day. Scheduling is triggered at 00:00 but not at 05:00.
*Scheduling Period	The task can be scheduled by hour or minute.
*Interval	Select a proper scheduling interval based on the selected scheduling period. The scheduling interval is the interval from the last running time. Manual synchronization is also counted in the running time. For example, if a task starts at 20:00 and manually executed at 20:03, and the interval is five minutes, the task is scheduled again at 20:08.

Step 5 After the user synchronization task is created, it does not run directly. You need to manually synchronize or schedule the task. The task takes effect after it is successfully synchronized. For details, see **Synchronizing or Scheduling Tasks**.

----End

Related Operations

- Synchronizing or scheduling tasks: On the user synchronization task page, click **Synchronize** or choose **More** > **Start Schedule** in the **Operation** column of the corresponding task. If a task has not been executed before and is scheduled for the first time, the task is triggered immediately.

  NOTE

  If a task fails to be executed, perform the following operations:

  - If the error message indicates insufficient permissions, see **Authorizing dlg_agency**.
  - If the error message indicates that the GaussDB(DWS) IAM credential fails to be downloaded, check whether the current user has at least the GaussDB(DWS) database access permission.
  - If error message "Mrs sync failed, please check the failure cause on the MRS page" is displayed for the MRS task, log in to the MRS console and choose **Operation Logs** in the navigation pane to view the cause.
  - If the MRS operation logs do not contain error information, the synchronization failure cause is that the IAM username conflicts with the name of an existing MRS human-machine user. Log in to MRS Manager and delete the human-machine user with the same name as the IAM user. The default description of the IAM username for the synchronization is **IAM Custom Policy User**, and the IAM user cannot be deleted. A common MRS human-machine user can be deleted.
  - Handle other errors based on the error messages and logs.

- Viewing task logs: On the user synchronization task page, locate the task whose logs need to be viewed and click **Details** in the **Operation** column to view the run logs. A maximum of 20 logs are displayed.

If a task fails to be executed, you can locate the failure cause based on logs, rectify the fault, and try to execute the task again. If the fault persists, contact technical support for assistance.

- Editing a task: On the user synchronization task page, click **More** in the **Operation** column and select **Edit** to edit the user synchronization task.
- Deleting a task: On the user synchronization task page, click **More** in the **Operation** column and select **Delete** to delete the user synchronization task. To delete multiple tasks at a time, select the tasks and click **Delete** above the task list.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.

9.3.5 Controlling Data Access Using Permissions

9.3.5.1 Configuring Workspace Permission Sets

In data access permission management, permissions are usually classified into multiple levels of permissions, such as those for level-1, level-2, and level-3 departments. DataArts Security provides a top-down hierarchical mode for data permission management. You can configure the maximum permissions in the workspace through a workspace permission set. Then, you can split the workspace permission set into permission sets for refined permission management.

A workspace permission set contains all the permissions for users in a DataArts Studio workspace. This permission set is created by the DAYU Administrator, Tenant Administrator, or data security administrator. A permission set contains only part of the permissions in a workspace.

Both a workspace permission set and a permission set directly associate users with permissions, but they differ in the following aspects:

- A workspace permission set is a top-level permission set that has no parent permission set. Generally, you only need to create one workspace permission set for each workspace. However, a permission set must be associated with a parent permission set, which can be a workspace permission set or another permission set. You can create multiple permission sets to associate users with different permissions in different scenarios.
- A workspace permission set mainly determines the permissions of a workspace, while a permission set is mainly used to manage permissions. A workspace permission set does not require permission synchronization and cannot be associated with roles. A permission set supports permission synchronization, which can be used for permission management, though associating a permission set with roles for permission management is more recommended.

This section describes how to **create** and **configure** a workspace permission set to define the permissions for a workspace.

Prerequisites

- A DWS connection, DLI connection, MRS Hive connection, and MRS Ranger connection have been created in Management Center based on **Creating a Data Connection**.
- Permissions have been configured for the **dlg_agency** based on **Authorizing dlg_agency**.
- User information has been synchronized from IAM to the data source based on **Synchronizing IAM Users to the Data Source**.
- Metadata of tables has been collected in DataArts Catalog through a **metadata collection task** if you want to view the metadata of databases, tables, and fields in data connections during permission configuration.

Constraints

- Only the DAYU Administrator, Tenant Administrator, or security administrator can create, modify, or synchronize workspace permission sets. The permission set administrator can synchronize workspace permission sets. Other common users cannot perform these operations.
- Workspace permission sets can only be used to define permissions for MRS Hive, DLI, and GaussDB(DWS).
- After a workspace permission set is configured, permission management does not take effect immediately. Instead, you need to synchronize the workspace permission set to the data source for permission management to take effect.

  Because workspace permission sets are mainly used to determine the permissions of workspaces rather than manage permissions, generally workspace permission sets do not need to be synchronized. You are advised to configure roles based on **Configuring Roles** to manage permissions. If you need to synchronize workspace permission sets, pay attention to the following restrictions:

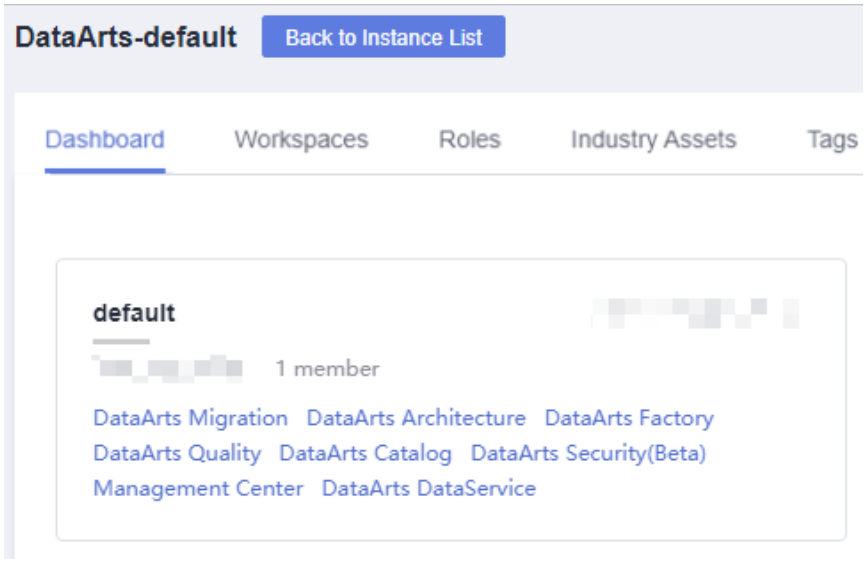
  - During authorization, the name of the object to be authorized (database, table, or column name) can contain only digits, letters, underscores (_), hyphens (-), and wildcards (*).
  - During DLI permission set synchronization, the custom policies created in IAM are associated with users or user groups. A maximum of 200 custom policies can be created in IAM. Before synchronization, ensure that the quotas are sufficient.
  - During permission synchronization, you need to configure required permissions for the **dlg_agency**. For details, see **Authorizing dlg_agency**.
- The current data permission control uses the allowlist mechanism, which adds operation conditions to the users to be authorized without affecting the permissions the users already have. If you only want to make the permissions granted by the data permission control take effect, you need to revoke the original permissions of the users to be authorized. For details, see **Data Permission Management**.
- Deleted workspace permission sets are moved to the recycle bin. You can restore them within 30 days. After 30 days, they will be deleted permanently. For details, see **Managing the Recycle Bin**.
- During script execution and job testing in DataArts Factory, the MRS or GaussDB(DWS) data source uses the account of the data connection for

authentication by default. Therefore, permission management still does not take effect during data development. You need to enable fine-grained authentication so that the current user is used for authentication during script execution and job testing in DataArts Factory. In this way, different users have different data permissions, and permission management for roles and permission sets takes effect.

Creating a Workspace Permission Set

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

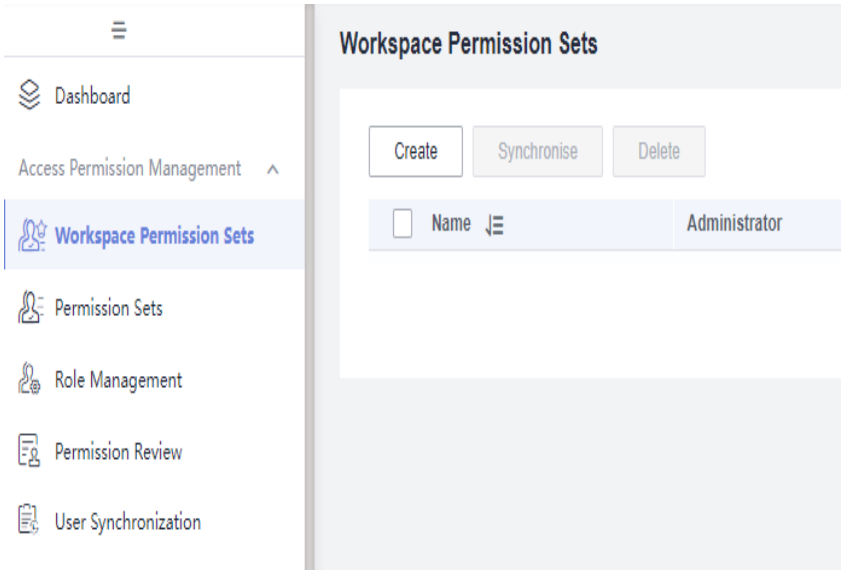
Figure 9-29 DataArts Security



Step 2 In the left navigation pane, choose **Workspace Permission Sets**.

Step 3 On the displayed page, click **Create**.

Figure 9-30 Creating a workspace permission set

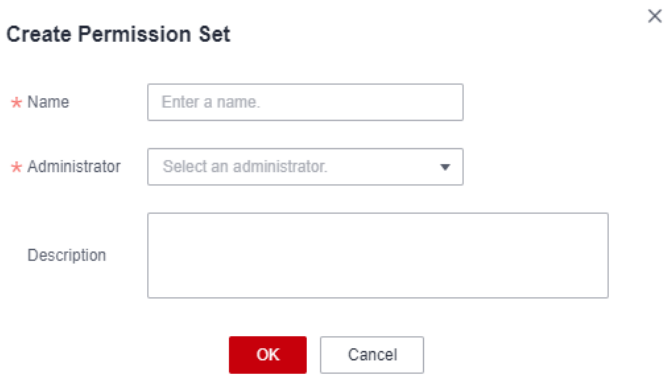


Step 4 Configure parameters based on **Table 9-4** and click **OK**.

Table 9-4 Parameters for creating a workspace permission set

Parameter	Description
*Name	Permission set name, which is unique in the instance. You should include the meaning of the permission set and avoid meaningless descriptions in the name so that the permission set can be quickly identified.
*Administrator	Select one or two administrators of the user or user group type. The administrators are the owners of the permission set and can configure the permissions in the permission set. The administrators can perform the following operations: • Permission configuration: Assign data source permissions to the workspace permission set. • User configuration: Assign permissions in the workspace permission set to users, user groups, or workspace roles. • Permission set creation: Create permission sets and roles based on the workspace permission set. The created permission sets do not contain more permissions than the workspace permission set.
Description	Information to make the workspace permission set easier to be identified

Figure 9-31 Creating a workspace permission set

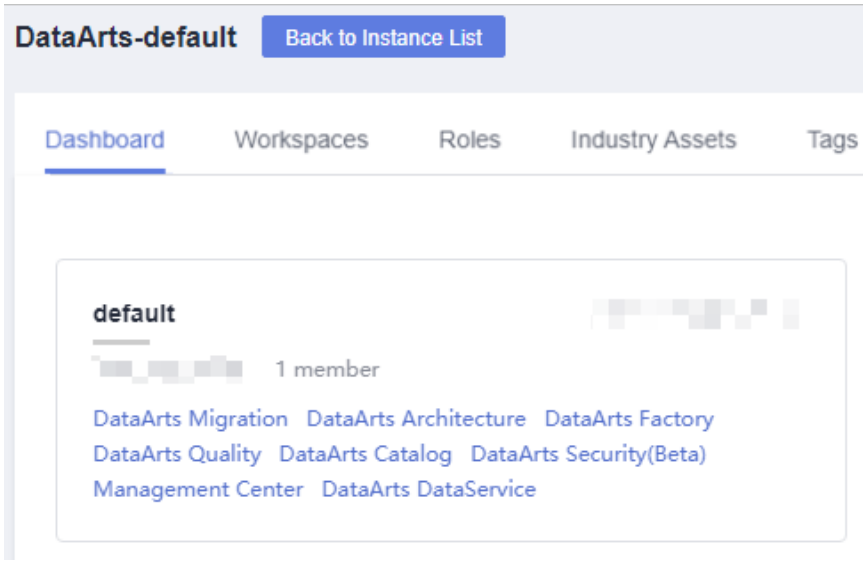


----End

Configuring the Workspace Permission Set

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

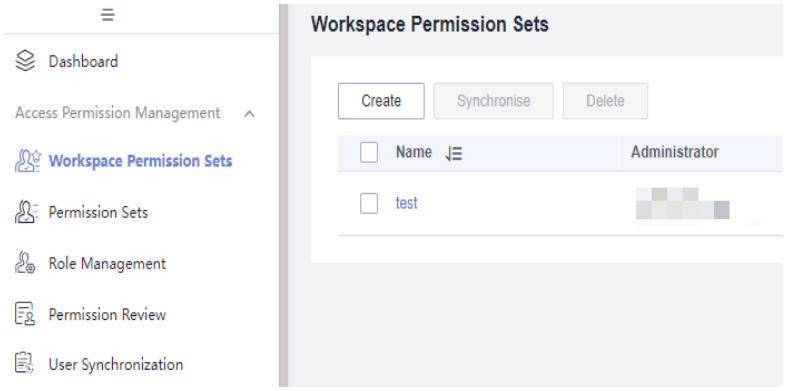
Figure 9-32 DataArts Security



Step 2 In the left navigation pane, choose **Workspace Permission Sets**.

Step 3 Locate a workspace permission set and click its name to go to the details page.

Figure 9-33 Going to the workspace permission set details page



Step 4 In the **Basic Information** area, you can view the name, ID, and administrator of the workspace permission set. For details, see **Figure 9-34**.

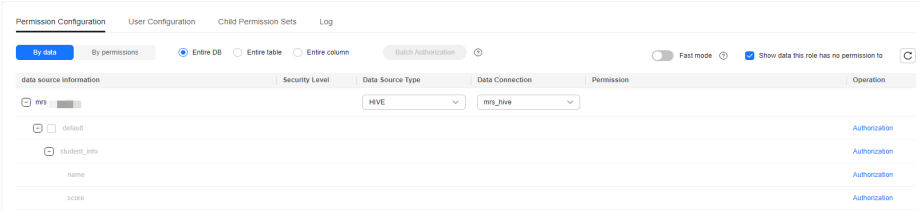
Figure 9-34 Basic information about the workspace permission set

Step 5 On the **Permission Configuration** tab page, **By data** is selected by default. You can select **By permissions**. The configured permissions are the same for **By data**

and **By permissions**, and the only difference lies in how the permissions are displayed. You are advised to select **By permissions** for batch authorization.

- **By data**: The system allows you to configure permissions for data. Currently, only MRS data sources are supported.

  Figure 9-35 Configuring permissions on the By data page

  When configuring permissions, you can select **Entire DB**, **Entire table**, or **Entire column**, and select the corresponding levels in the data source information to perform a batch authorization. You can also click **Authorization** in the **Operation** column of a data record in the expanded navigation pane to authorize access to the data.

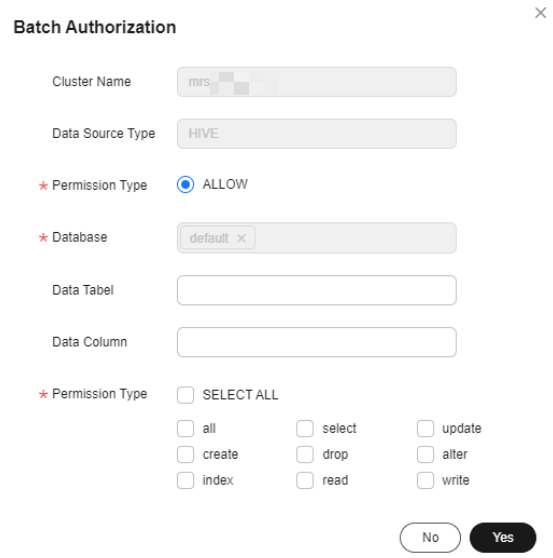
  Fast mode and **Show data this role has no permission to** are supported. If **Fast mode** is enabled, metadata of databases, tables, and columns is obtained from DataArts Catalog. Otherwise, metadata is obtained from the data source. If metadata has been collected, you are advised to enable **Fast mode**.

  NOTE

  - Note that the permissions of databases, tables, and columns are managed by layer. For example, a user who has been granted database permissions does not have the permissions of tables and columns. Table and column permissions must be granted separately.

    For example, if you enter a table name or an asterisk (*) as a wildcard during database authorization, you are authorizing the table. If you enter a column name or an asterisk (*) as a wildcard character, you are authorizing the column.
  - During authorization, the name of the object to be authorized (database, table, or column name) can contain only digits, letters, underscores (_), hyphens (-), and wildcards (*).

Figure 9-36 Authorization on the By data page



- **By permissions**: The system allows you to configure permissions.

  To configure permissions, click **Add** and select data levels in sequence. You cannot select multiple objects at the same level (such as database, table, and column) for batch authorization. **Permission Type** cannot be set to **DENY**.

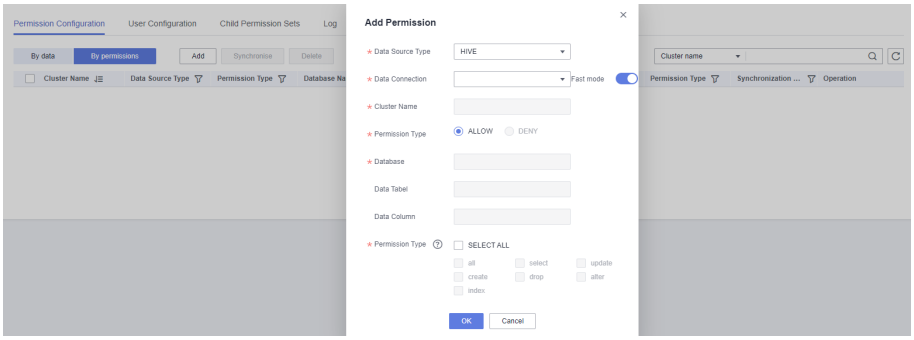
  NOTE

  - Note that the permissions of databases, tables, and columns are managed by layer. For example, a user who has been granted database permissions does not have the permissions of tables and columns. Table and column permissions must be granted separately.

    For example, if you enter a table name or an asterisk (*) as a wildcard during database authorization, you are authorizing the table. If you enter a column name or an asterisk (*) as a wildcard character, you are authorizing the column.
  - During authorization, the name of the object to be authorized (database, table, or column name) can contain only digits, letters, underscores (_), hyphens (-), and wildcards (*).

  After configuring permissions, you can edit, synchronize, or delete them.

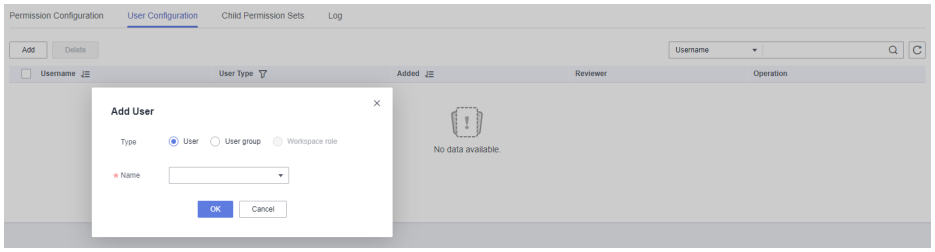
Figure 9-37 Configuring permissions on the By permissions page



Step 6 **User Configuration**: On the permission set details page, click the **User Configuration** tab.

On this page, you can associate the permissions configured on the **Permission Configuration** page with users. Click **Add** and select **User** or **User group** (**Workspace role** is unavailable currently) to add users to the permission set. You can select users or user groups that have been added to the workspace.

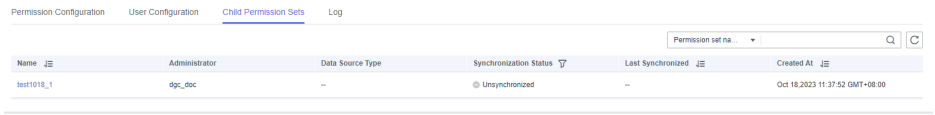
Figure 9-38 User Configuration



Step 7 **Child Permission Sets**: On the permission set details page, click the **Child Permission Sets** tab.

On this page, you can view the child permission sets of the current permission set.

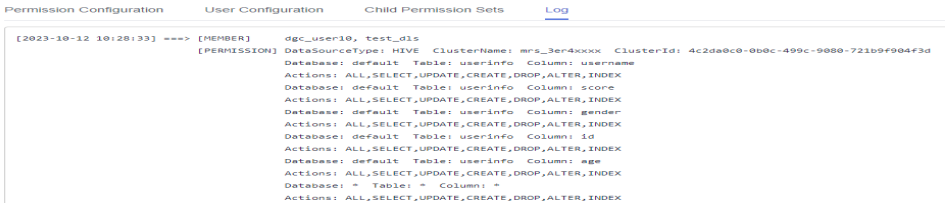
Figure 9-39 View child permission sets



Step 8 **Log**: On the permission set details page, click the **Log** tab.

On this page, you can view the log details if permission synchronization fails. The system deletes logs generated 30 days ago at 00:00 every day.

Figure 9-40 Viewing logs



Step 9 After the permission set is configured, permission management does not take effect immediately. You need to manually synchronize permissions to the data source for permission management to take effect. For details, see **Synchronizing Permission Sets**.

Because workspace permission sets are mainly used to determine the permissions of workspaces rather than manage permissions, generally workspace permission sets do not need to be synchronized. You are advised to configure roles based on **Configuring Roles** to manage permissions.

----End

Related Operations

- Synchronizing workspace permission sets: Workspace permission sets take effect only after they are manually synchronized to the data source. Because workspace permission sets are mainly used to determine the permissions of workspaces rather than manage permissions, generally workspace permission sets do not need to be synchronized. You are advised to configure roles based on **Configuring Roles** to manage permissions.

  To synchronize a workspace permission set, click **Synchronize** in the **Operation** column of the permission set on the **Workspace Permission Sets** page. To synchronize multiple permission sets, select them and click **Synchronize** above the list.
- Editing a workspace permission set: On the **Workspace Permission Sets** page, click **Edit** in the **Operation** column of a permission set. You can change the name, administrator, and description of the permission set.
- Deleting workspace permission sets: On the **Workspace Permission Sets** page, click **Delete** in the **Operation** column of a permission set. In the displayed dialog box, confirm the permission set to delete and click **Yes**. To delete multiple permission sets, select them and click **Delete** above the list.

  Workspace permission sets for which permissions, users, or child permission sets have been configured cannot be deleted. To delete such workspace permission sets, delete the configurations first.

  NOTE

  Deleted workspace permission sets are moved to the recycle bin. You can restore them within 30 days. After 30 days, they will be deleted permanently. For details, see **Managing the Recycle Bin**.

9.3.5.2 Configuring Permission Sets

In data access permission management, permissions are usually classified into multiple levels of permissions, such as those for level-1, level-2, and level-3 departments. DataArts Security provides a top-down hierarchical mode for data permission management. You can configure the maximum permissions in the workspace through a workspace permission set. Then, you can split the workspace permission set into permission sets for refined permission management.

A permission set directly associates users and permissions. A workspace permission set is special as it has no parent permission set. It defines the permissions for the entire workspace. Each child permission set defined in the workspace permission set has a parent permission set, and the permissions of a child permission set are a subset of its parent permission set's permissions.

Both a workspace permission set and a permission set directly associate users with permissions, but they differ in the following aspects:

- A workspace permission set is a top-level permission set that has no parent permission set. Generally, you only need to create one workspace permission set for each workspace. However, a permission set must be associated with a parent permission set, which can be a workspace permission set or another permission set. You can create multiple permission sets to associate users with different permissions in different scenarios.
- A workspace permission set mainly determines the permissions of a workspace, while a permission set is mainly used to manage permissions. A

workspace permission set does not require permission synchronization and cannot be associated with roles. A permission set supports permission synchronization, which can be used for permission management, though associating a permission set with roles for permission management is more recommended.

This section describes how to manage permissions through **Creating a Permission Set** and **Configuring the Permission Set**. In practice, you are advised to manage permissions based on **Configuring Roles**.

Prerequisites

- You have configured a workspace permission set. For details, see **Configuring Workspace Permission Sets**.
- Metadata of tables has been collected in DataArts Catalog through a **metadata collection task** if you want to view the metadata of databases, tables, and fields in data connections during permission configuration.

Constraints

- Only the DAYU Administrator, Tenant Administrator, data security administrator, and the administrator of the parent permission set can create, modify, and synchronize permission sets. The permission set administrator can synchronize workspace permission sets. Other common users cannot perform these operations.
- Permission sets can only be used to manage permissions for MRS Hive, DLI, and GaussDB(DWS).
- In some cases, a child permission set may contain more permissions than its parent permission set. For example, this may occur if a permission record is configured for a child permission set and then deleted from the parent permission set, because cascading deletion of permissions is not supported.
- After a permission set is configured, permission management does not take effect immediately. Instead, you need to synchronize the permission set to the data source for permission management to take effect.

  Role management provides more intuitive and powerful permission management capabilities based on permission sets. Generally, you do not need to synchronize permission sets except for DLI data sources. You are advised to manage permissions based on **Configuring Roles**. If you need to synchronize workspace permission sets, pay attention to the following restrictions:

  - During authorization, the name of the object to be authorized (database, table, or column name) can contain only digits, letters, underscores (_), hyphens (-), and wildcards (*).
  - During DLI permission set synchronization, the custom policies created in IAM are associated with users or user groups. A maximum of 200 custom policies can be created in IAM. Before synchronization, ensure that the quotas are sufficient.
  - During permission synchronization, you need to configure required permissions for the **dlg_agency**. For details, see **Authorizing dlg_agency**.
- The current data permission control uses the allowlist mechanism, which adds operation conditions to the users to be authorized without affecting the

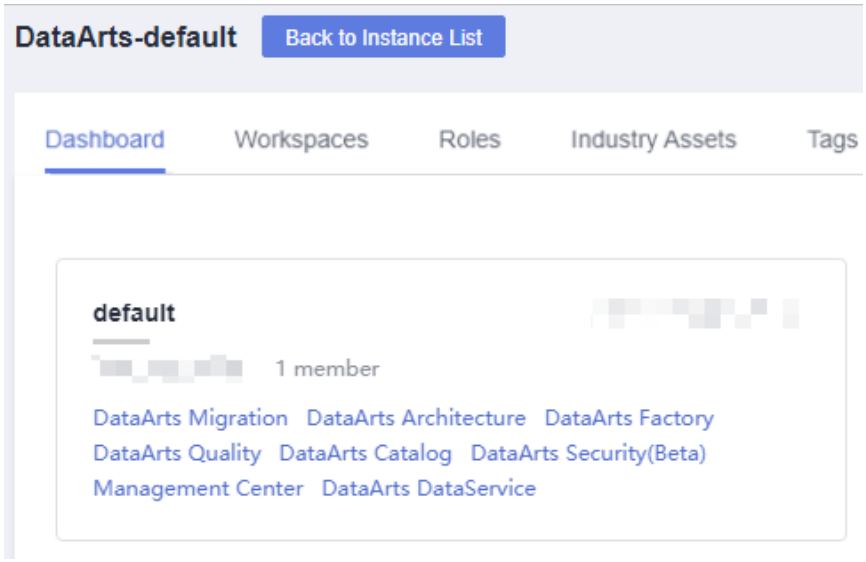
permissions the users already have. If you only want to make the permissions granted by the data permission control take effect, you need to revoke the original permissions of the users to be authorized. For details, see **Data Permission Management**.

- During script execution and job testing in DataArts Factory, the MRS or GaussDB(DWS) data source uses the account of the data connection for authentication by default. Therefore, permission management still does not take effect during data development. You need to enable fine-grained authentication so that the current user is used for authentication during script execution and job testing in DataArts Factory. In this way, different users have different data permissions, and permission management for roles and permission sets takes effect.

Creating a Permission Set

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

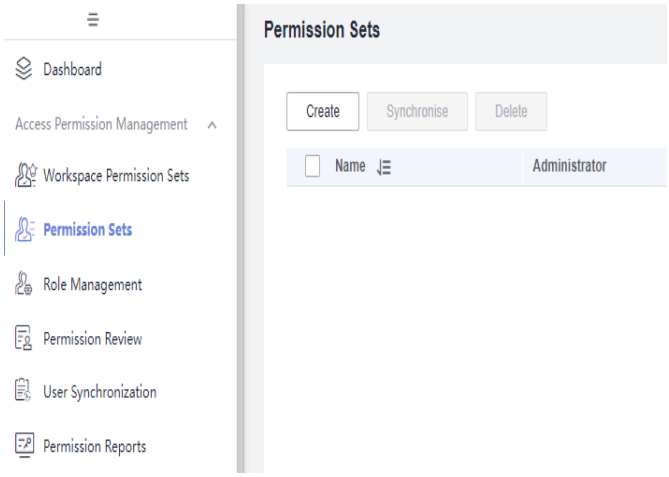
Figure 9-41 DataArts Security



Step 2 In the left navigation pane, choose **Permission Sets**.

Step 3 On the displayed page, click **Create**.

Figure 9-42 Creating a permission set

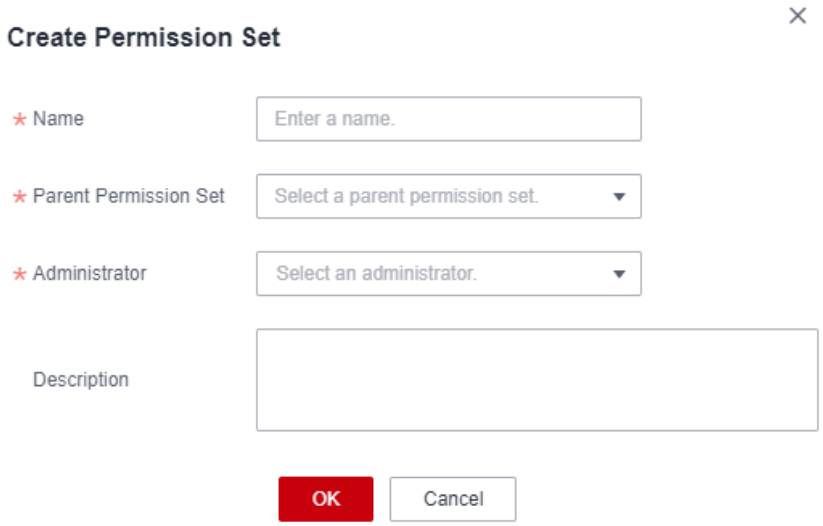


Step 4 Configure parameters based on **Table 9-5** and click **OK**.

Table 9-5 Parameters

Parameter	Description
*Name	Permission set name, which is unique in the instance. You should include the meaning of the permission set and avoid meaningless descriptions in the name so that the permission set can be quickly identified.
*Parent Permission Set	Select a parent permission set, which can be a workspace permission set or another permission set. After you select a parent permission set, the permissions of the current permission set are a subset of the parent permission set's permissions.
*Administrator	The administrators are the owners of the permission set and can configure the permissions in the permission set. The administrators can perform the following operations: • Permission configuration: Assign data source permissions to the workspace permission set. • User configuration: Assign permissions in the workspace permission set to users, user groups, or workspace roles. • Permission set creation: Create permission sets and roles based on the workspace permission set. The created permission sets do not contain more permissions than the workspace permission set.
Description	Information to make the permission set easier to be identified

Figure 9-43 Parameters for creating a permission set

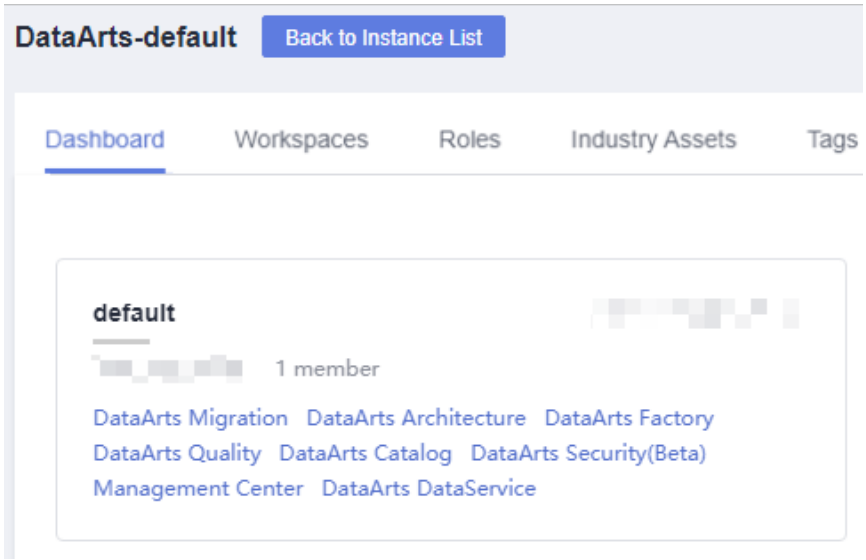


----End

Configuring the Permission Set

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

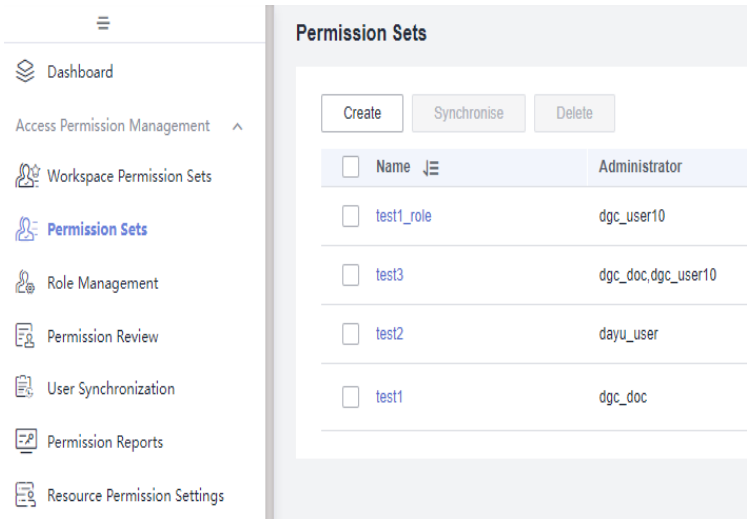
Figure 9-44 DataArts Security



Step 2 In the left navigation pane, choose **Permission Sets**.

Step 3 Locate a permission set and click its name to go to the details page.

Figure 9-45 Going to the permission set details page



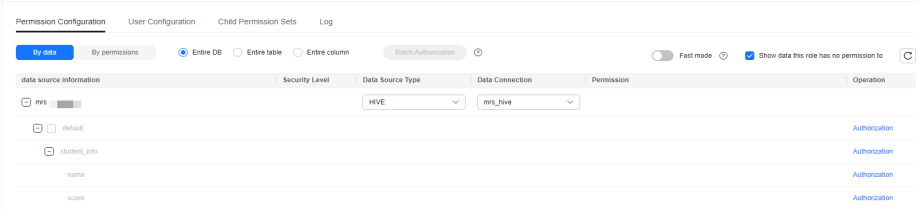
Step 4 In the **Basic Information** area, you can view the name, ID, and administrator of the permission set. For details, see **Figure 9-46**.

Figure 9-46 Basic information of the permission set

Step 5 On the **Permission Configuration** tab page, **By data** is selected by default. You can select **By permissions**. The configured permissions are the same for **By data** and **By permissions**, and the only difference lies in how the permissions are displayed. You are advised to select **By permissions** for batch authorization.

- **By data**: The system allows you to configure permissions for data. (Currently, only MRS data sources are supported.) You can select the authorized data in the parent permission set.

  Figure 9-47 Configuring permissions on the By data page

  When configuring permissions, you can select **Entire DB**, **Entire table**, or **Entire column**, and select the corresponding levels in the data source information to perform a batch authorization. You can also click **Authorization** in the **Operation** column of a data record in the expanded navigation pane to authorize access to the data.

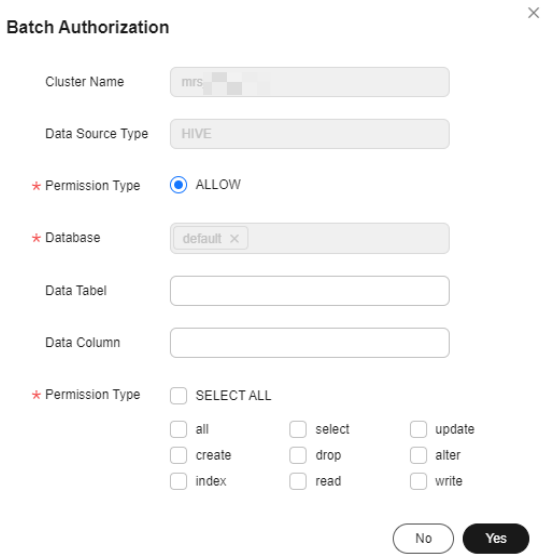
Fast mode and **Show data this role has no permission to** are supported. If **Fast mode** is enabled, metadata of databases, tables, and columns is obtained from DataArts Catalog. Otherwise, metadata is obtained from the data source. If metadata has been collected, you are advised to enable **Fast mode**.

NOTE

- Note that the permissions of databases, tables, and columns are managed by layer. For example, a user who has been granted database permissions does not have the permissions of tables and columns. Table and column permissions must be granted separately.

  For example, if you enter a table name or an asterisk (*) as a wildcard during database authorization, you are authorizing the table. If you enter a column name or an asterisk (*) as a wildcard character, you are authorizing the column.
- During authorization, the name of the object to be authorized (database, table, or column name) can contain only digits, letters, underscores (_), hyphens (-), and wildcards (*).

Figure 9-48 Authorization on the By data page



- **By permissions**: The system allows you to configure permissions. You can select the authorized data in the parent permission set.

  To configure permissions, click **Add** and select data levels in sequence. You cannot select multiple objects at the same level (such as database, table, and column) for batch authorization. **Permission Type** cannot be set to **DENY**.

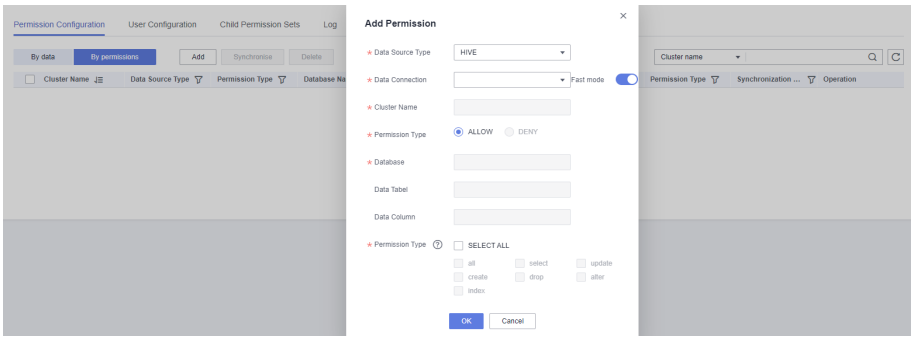
  NOTE

  - Note that the permissions of databases, tables, and columns are managed by layer. For example, a user who has been granted database permissions does not have the permissions of tables and columns. Table and column permissions must be granted separately.

    For example, if you enter a table name or an asterisk (*) as a wildcard during database authorization, you are authorizing the table. If you enter a column name or an asterisk (*) as a wildcard character, you are authorizing the column.
  - During authorization, the name of the object to be authorized (database, table, or column name) can contain only digits, letters, underscores (_), hyphens (-), and wildcards (*).

After configuring permissions, you can edit, synchronize, or delete them.

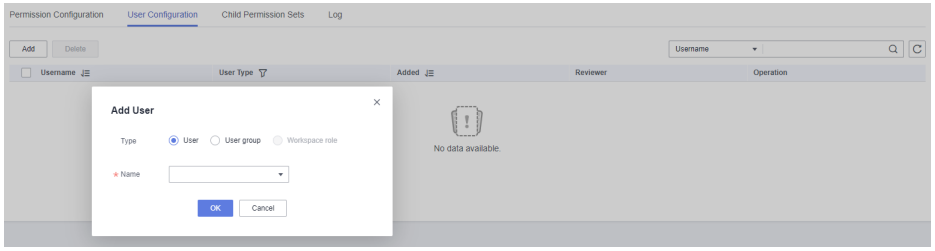
Figure 9-49 Configuring permissions on the By permissions page



Step 6 **User Configuration**: On the permission set details page, click the **User Configuration** tab.

On this page, you can associate the permissions configured on the **Permission Configuration** page with users. Click **Add** and select **User** or **User group** (**Workspace role** is unavailable currently) to add users to the permission set. You can select users or user groups that have been added to the workspace.

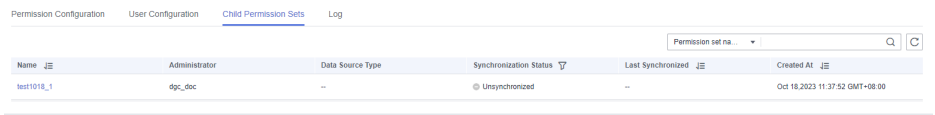
Figure 9-50 User Configuration



Step 7 **Child Permission Sets**: On the permission set details page, click the **Child Permission Sets** tab.

On this page, you can view the child permission sets of the current permission set.

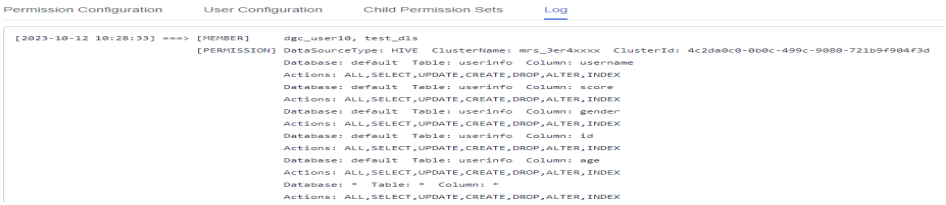
Figure 9-51 View child permission sets



Step 8 **Log**: On the permission set details page, click the **Log** tab.

On this page, you can view the log details if permission synchronization fails. The system deletes logs generated 30 days ago at 00:00 every day.

Figure 9-52 Viewing logs



Step 9 After the permission set is configured, it does not take effect immediately. You need to manually synchronize the permission set to the data source for permission management to take effect. For details, see **Synchronizing Permission Sets**.

Role management provides more intuitive and powerful permission management capabilities based on permission sets. Generally, you do not need to synchronize workspace permission sets. In practice, you are advised to manage permissions based on **Configuring Roles**.

----End

Related Operations

- Synchronizing permission sets: Permission sets take effect only after they are synchronized to the data source. Role management provides more intuitive and powerful permission management capabilities based on permission sets. Generally, you do not need to synchronize permission sets. In practice, you are advised to manage permissions based on **Configuring Roles**.

  To synchronize a permission set, click **Synchronize** in the **Operation** column of the permission set on the **Permission Sets** page. To synchronize multiple permission sets, select them and click **Synchronize** above the list.
- Editing a permission set: On the **Permission Sets** page, click **Edit** in the **Operation** column of a permission set. You can change the name, administrator, and description of the permission set.
- Deleting permission sets: On the **Permission Sets** page, click **Delete** in the **Operation** column of a permission set. In the displayed dialog box, confirm the permission set to delete and click **Yes**. To delete multiple permission sets, select them and click **Delete** above the list.

  Permission sets for which permissions, users, or child permission sets have been configured cannot be deleted. To delete such permission sets, delete the configurations first.

  NOTE

  Deleted permission sets are moved to the recycle bin. You can restore them within 30 days. After 30 days, they will be deleted permanently. For details, see **Managing the Recycle Bin**.

9.3.5.3 Configuring Roles

Role management in DataArts Security provides more intuitive and powerful permission management capabilities based on permission sets. The difference between a role and a permission set is that a permission set directly associates users with permissions, while a role is created or managed on the data source to carry the association between users and permissions.

If you associate roles with permission sets on the role management page, permissions are synchronized only to roles instead of users. You are advised to use role management to manage permissions and permission relationships more intuitively. Role management also allows you to use managed roles to manage existing data source permissions.

- Common roles: Create roles on the data source to associate users and permissions.
- Manage roles: Manage existing roles on the MRS data source and inherit their permissions of the MRS data source. (To view existing roles on the MRS data source, log in to MRS FusionInsight Manager and choose **System** > **Permission** > **Role**).

This section describes **Configuring a Common Role**, **Configuring Managed Roles**, and **Related Operations**.

Prerequisites

- You have configured a workspace permission set. For details, see **Configuring Workspace Permission Sets**.
- During the synchronization of MRS and GaussDB(DWS) roles, the system uses the users in the data connections in Management Center to perform addition, deletion, modification, and query operations. Users in the data connections must have the following permissions:
  - Users in MRS Ranger connections must have the admin permission of the Ranger component.
  - In non-rights separation mode (RSM), database users in GaussDB(DWS) connections must have at least the dbadmin permission of the database. In RSM, users must have the system administrator permissions.

  For details about the configuration method, see **Checking the Cluster Version and Permissions**.
- Metadata of tables has been collected in DataArts Catalog through a **metadata collection task** if you want to view the metadata of databases, tables, and fields in data connections during permission configuration in fast mode.

Constraints

- Currently, roles can only be created for MRS and GaussDB(DWS) clusters.
- Workspace permission sets are mainly used to define the permissions of workspaces rather than manage permissions. Roles cannot be created for workspace permission sets.
- During authorization, the name of the object to be authorized (database, table, or column name) can contain only digits, letters, underscores (_), hyphens (-), and wildcards (*).
- If you create roles for permission sets, permissions are synchronized only to roles instead of users.
- Role management is available only when the version of the CDM cluster selected for the agent in the data connection is 2.10.0.300 or later.
- During the synchronization of MRS and GaussDB(DWS) roles, the system uses the users in the data connections in Management Center to perform addition,

deletion, modification, and query operations. Users in the data connections must have the following permissions:

- Users in MRS Ranger connections must have the admin permission of the Ranger component.
- In non-rights separation mode (RSM), database users in GaussDB(DWS) connections must have at least the dbadmin permission of the database. In RSM, users must have the system administrator permissions.

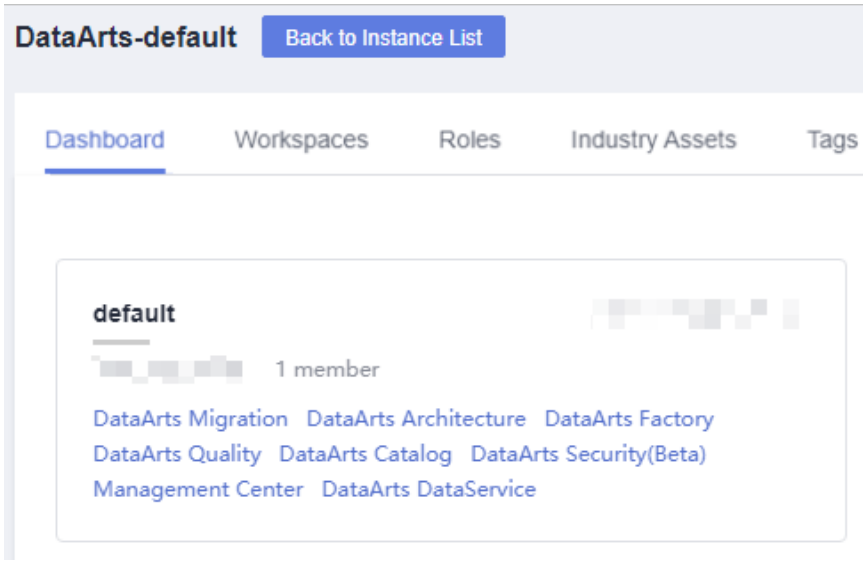
For details about the configuration method, see **Checking the Cluster Version and Permissions**.

- Only the directory permissions of the cluster are displayed for roles in the workspace.
- During permission synchronization, you need to configure required permissions for the **dlg_agency**. For details, see **Authorizing dlg_agency**.
- The current data permission control uses the allowlist mechanism, which adds operation conditions to the users to be authorized without affecting the permissions the users already have. If you only want to make the permissions granted by the data permission control take effect, you need to revoke the original permissions of the users to be authorized. For details, see **Data Permission Management**.
- During script execution and job testing in DataArts Factory, the MRS or GaussDB(DWS) data source uses the account of the data connection for authentication by default. Therefore, permission management still does not take effect during data development. You need to enable fine-grained authentication so that the current user is used for authentication during script execution and job testing in DataArts Factory. In this way, different users have different data permissions, and permission management for roles and permission sets takes effect.

Configuring a Common Role

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-53 DataArts Security

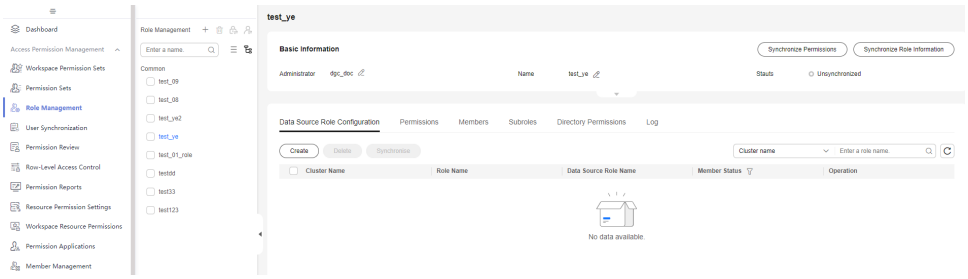


Step 2 In the navigation pane on the left, choose **Role Management**.

Step 3 Use either of the following methods to configure a common role:

- Configuring an existing role: On the **Role Management** page, permission sets that have been created in **Creating a Permission Set** are displayed in the navigation tree as common roles by default. You can click a role name to go to the role details page.

  Figure 9-54 Role details page

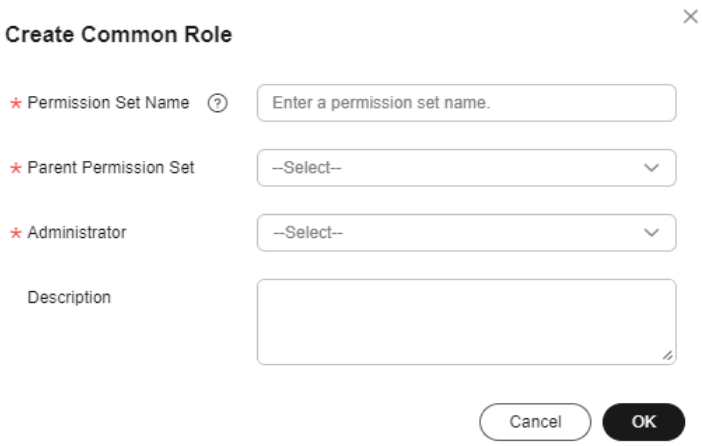
- Creating a role: On the **Role Management** page, click + in the navigation tree and select **Create Common Role**. Set the parameters listed in **Table 9-6** and click **OK**. The details page of the created role is displayed by default.

  Table 9-6 Parameters

Parameter	Description
*Name	Permission set name, which is unique in the instance. You should include the meaning of the permission set and avoid meaningless descriptions in the name so that the permission set can be quickly identified.
*Parent Permission Set	Select a parent permission set, which can be a workspace permission set or another permission set. After you select a parent permission set, the permissions of the current permission set are a subset of the parent permission set's permissions.
*Administrator	The administrators are the owners of the permission set and can configure the permissions in the permission set. The administrators can perform the following operations: – Permission configuration: Assign data source permissions to the workspace permission set. – User configuration: Assign permissions in the workspace permission set to users, user groups, or workspace roles. – Permission set creation: Create permission sets and roles based on the workspace permission set. The created permission sets do not contain more permissions than the workspace permission set.

Parameter	Description
Description	Information to make the permission set easier to be identified

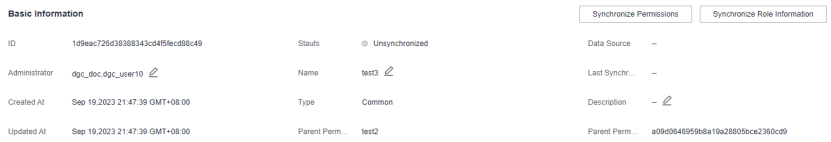
Figure 9-55 Creating a common role



Step 4 On the role details page, you can expand the **Basic Information** area to view the name, ID, and administrator of the role. For details, see **Figure 9-56**.

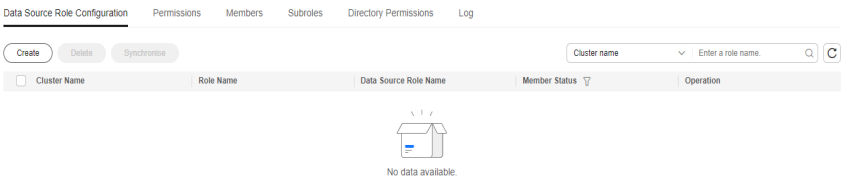
After configuring roles and permissions, you can synchronize them by clicking **Synchronize Permissions** and **Synchronize Role Information** in the upper right corner.

Figure 9-56 Basic role information



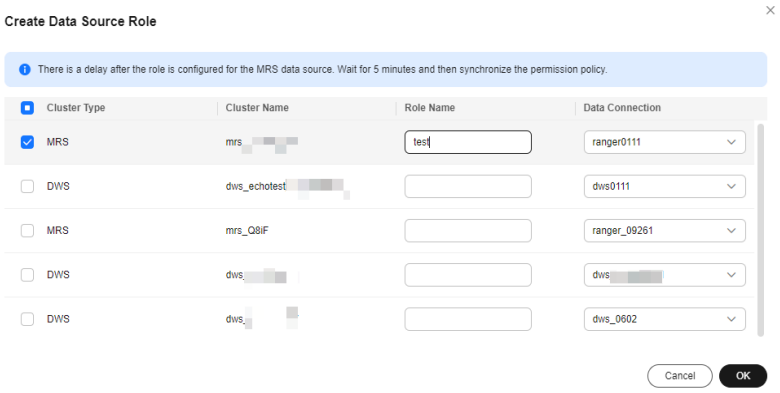
Step 5 **Data Source Role Configuration**: On this page, you can click **Create** to create roles for associating users and permissions.

Figure 9-57 Data Source Role Configuration page



Click **Create**. In the displayed dialog box, select data sources, set **Role Name**, and click **OK**.

Figure 9-58 Creating a data source role

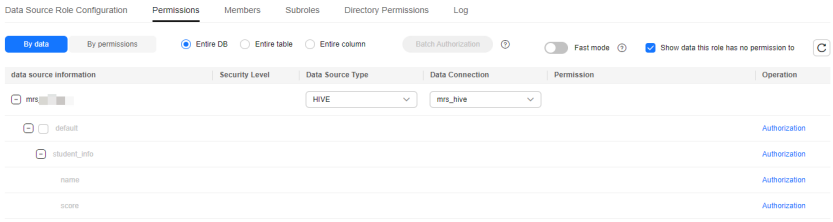


If you no longer need a data source role, click **Delete** in the **Operation** column to delete the role. After the role is deleted, permissions are no longer synchronized to the role and only synchronized to user information.

Step 6 **Permissions**: On the role details page, click the **Permissions** tab. By default, **By data** is selected. You can also select **By permissions**. The configured permissions are the same for **By data** and **By permissions**, and the only difference lies in how the permissions are displayed. You are advised to select **By permissions** for batch authorization.

- **By data**: The system allows you to configure permissions for data. Currently, only MRS data sources are supported.

  Figure 9-59 Configuring permissions on the By data page

  When configuring permissions, you can select **Entire DB**, **Entire table**, or **Entire column**, and select the corresponding levels in the data source information to perform a batch authorization. You can also click **Authorization** in the **Operation** column of a data record in the expanded navigation pane to authorize access to the data.

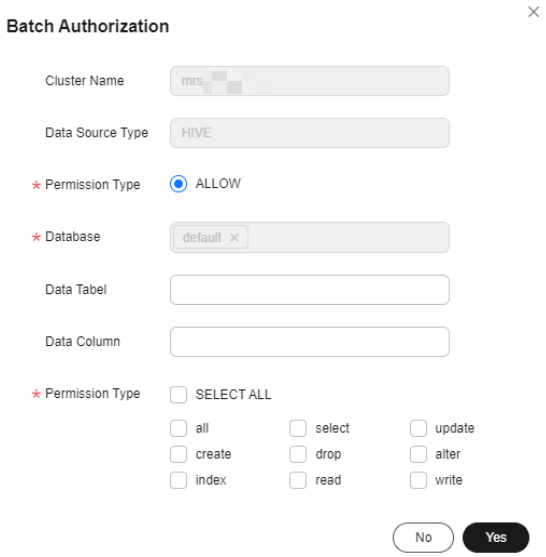
  Fast mode and **Show data this role has no permission to** are supported. If **Fast mode** is enabled, metadata of databases, tables, and columns is obtained from DataArts Catalog. Otherwise, metadata is obtained from the data source. If metadata has been collected, you are advised to enable **Fast mode**.

NOTE

- Note that the permissions of databases, tables, and columns are managed by layer. For example, a user who has been granted database permissions does not have the permissions of tables and columns. Table and column permissions must be granted separately.

  For example, if you enter a table name or an asterisk (*) as a wildcard during database authorization, you are authorizing the table. If you enter a column name or an asterisk (*) as a wildcard character, you are authorizing the column.
- During authorization, the name of the object to be authorized (database, table, or column name) can contain only digits, letters, underscores (_), hyphens (-), and wildcards (*).

Figure 9-60 Authorization on the By data page



- **By permissions**: The system allows you to configure permissions.

  To configure permissions, click **Add** and select data levels in sequence. You cannot select multiple objects at the same level (such as database, table, and column) for batch authorization. **Permission Type** cannot be set to **DENY**.

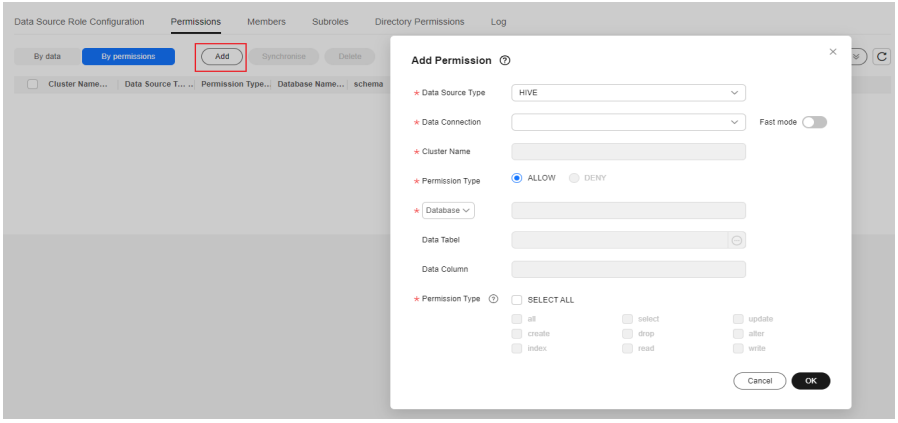
  NOTE

  - Note that the permissions of databases, tables, and columns are managed by layer. For example, a user who has been granted database permissions does not have the permissions of tables and columns. Table and column permissions must be granted separately.

    For example, if you enter a table name or an asterisk (*) as a wildcard during database authorization, you are authorizing the table. If you enter a column name or an asterisk (*) as a wildcard character, you are authorizing the column.
  - During authorization, the name of the object to be authorized (database, table, or column name) can contain only digits, letters, underscores (_), hyphens (-), and wildcards (*).

  After configuring permissions, you can edit, synchronize, or delete them.

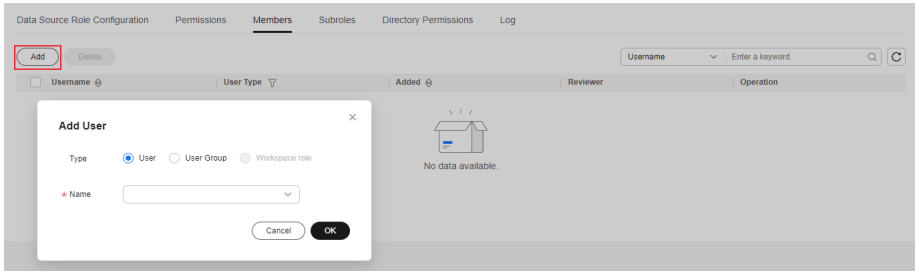
Figure 9-61 Configuring permissions on the By permissions page



Step 7 **Members**: On the role details page, click the **Members** tab.

Members associate the roles on the **Data Source Role Configuration** page with users. Click **Add** to add users, user groups, or workspace roles to roles. You can select users or user groups that have been added to the workspace.

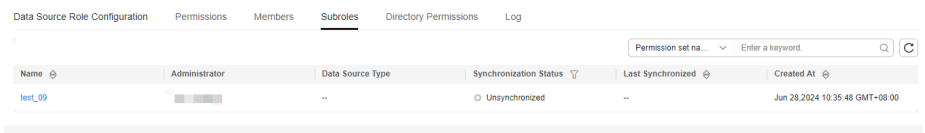
Figure 9-62 Members



Step 8 **Subroles**: On the role details page, click the **Subroles** tab.

On this page, you can view the subroles of the current role.

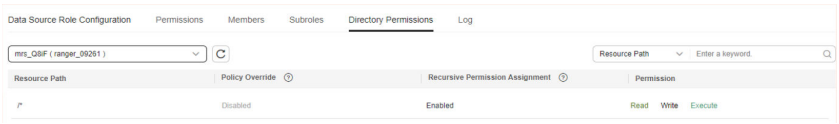
Figure 9-63 Viewing subroles



Step 9 **Directory Permissions**: On the role details page, click the **Directory Permissions** tab.

Directory permissions obtain the HDFS policies of this role from the Ranger component to display the HDFS paths to which this role has permissions. In addition, you can view the operation permissions of the paths. You can search for the permissions of a path. Only exact match is supported.

Figure 9-64 Viewing directory permissions



Step 10 **Log**: On the role details page, click the **Log** tab.

On this page, you can view the log details if permission synchronization fails. The system deletes logs generated 30 days ago at 00:00 every day.

Figure 9-65 Viewing logs

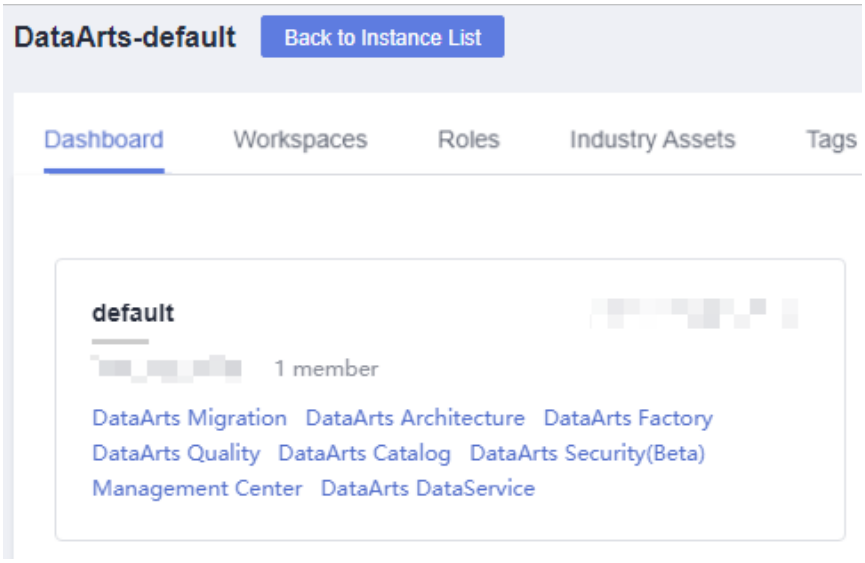
Step 11 After the role is configured, it does not take effect immediately. You need to synchronize the permissions and role to the data source for permission management to take effect. For details, see **Related Operations**.

----End

Configuring Managed Roles

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-66 DataArts Security

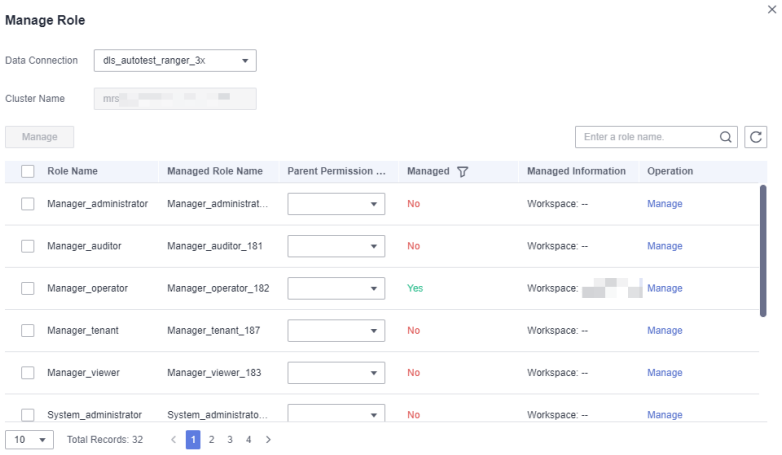


Step 2 In the navigation pane on the left, choose **Role Management**.

Step 3 On the **Role Management** page, click + in the navigation tree and select **Create Managed Role**. In the displayed dialog box, select a Ranger connection, set **Parent Permission Set/Role**, and click **Manage** in the **Operation** column of the MRS roles to be managed. You can also select multiple MRS roles to be managed and click **Manage** above the list.

If you no longer want to manage roles, you can delete the managed roles from the role management navigation tree. After the managed roles are deleted, permissions are no longer synchronized to the roles and only synchronized to user information.

Figure 9-67 Creating a managed role

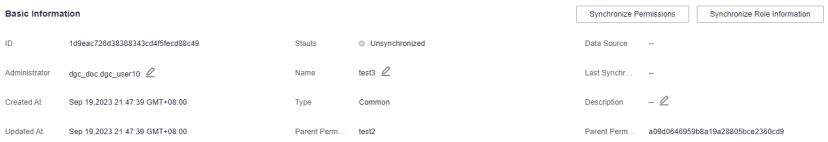


Step 4 Close the **Manage Role** dialog box and return to the **Role Management** page. In the role management navigation tree, locate the MRS role added in the previous step and click the role name to go to the role details page.

Step 5 On the role details page, you can expand the **Basic Information** area to view the name, ID, and administrator of the role. For details, see **Figure 9-68**.

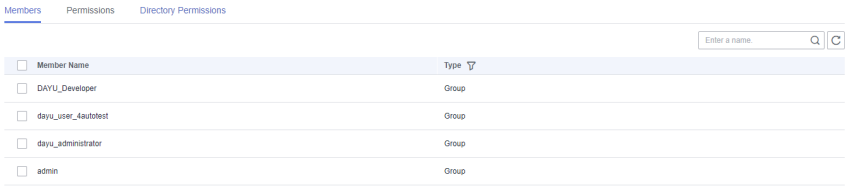
After configuring roles and permissions, you can synchronize them by clicking **Synchronize Permissions** and **Synchronize Role Information** in the upper right corner.

Figure 9-68 Basic role information



Step 6 **Members**: On this page, you can view the users or user groups associated with the MRS role. Currently, users cannot be added to managed roles in DataArts Security.

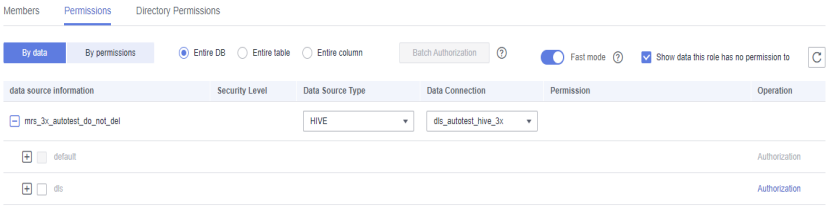
Figure 9-69 Members



Step 7 **Permissions**: On the role details page, click the **Permissions** tab. By default, **By data** is selected. You can also select **By permissions**. The configured permissions are the same for **By data** and **By permissions**, and the only difference lies in how the permissions are displayed. You are advised to select **By permissions** for batch authorization.

- **By data**: The system allows you to configure permissions. If a **metadata collection task** has been executed successfully, you can view the data source information and click to expand the navigation pane.

  Figure 9-70 Configuring permissions on the By data page

  When configuring permissions, you can select **Entire DB**, **Entire table**, or **Entire column**, and select the corresponding levels in the data source information to perform a batch authorization. You can also click **Authorization** in the **Operation** column of a data record in the expanded navigation pane to authorize access to the data.

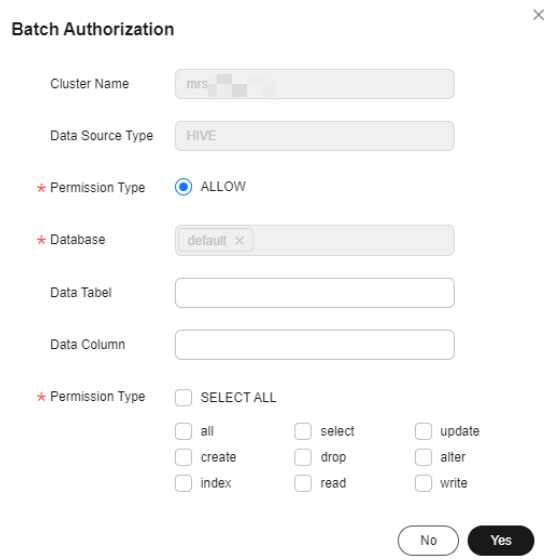
  Fast mode and **Show data this role has no permission to** are supported. If **Fast mode** is enabled, metadata of databases, tables, and columns is obtained from DataArts Catalog. Otherwise, metadata is obtained from the data source. If metadata has been collected, you are advised to enable **Fast mode**.

  NOTE

  - Note that the permissions of databases, tables, and columns are managed by layer. For example, a user who has been granted database permissions does not have the permissions of tables and columns. Table and column permissions must be granted separately.

    For example, if you enter a table name or an asterisk (*) as a wildcard during database authorization, you are authorizing the table. If you enter a column name or an asterisk (*) as a wildcard character, you are authorizing the column.
  - During authorization, the name of the object to be authorized (database, table, or column name) can contain only digits, letters, underscores (_), hyphens (-), and wildcards (*).

Figure 9-71 Authorization on the By data page



- **By permissions**: The system allows you to configure permissions.

  To configure permissions, click **Add** and select data levels in sequence. You cannot select multiple objects at the same level (such as database, table, and column) for batch authorization. **Permission Type** cannot be set to **DENY**.

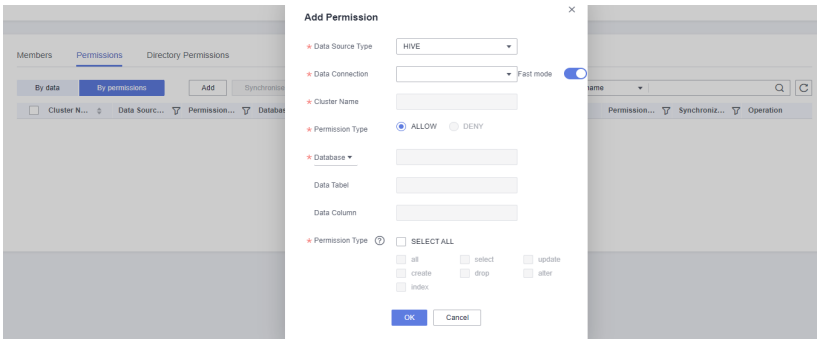
  NOTE

  - Note that the permissions of databases, tables, and columns are managed by layer. For example, a user who has been granted database permissions does not have the permissions of tables and columns. Table and column permissions must be granted separately.

    For example, if you enter a table name or an asterisk (*) as a wildcard during database authorization, you are authorizing the table. If you enter a column name or an asterisk (*) as a wildcard character, you are authorizing the column.
  - During authorization, the name of the object to be authorized (database, table, or column name) can contain only digits, letters, underscores (_), hyphens (-), and wildcards (*).

  After configuring permissions, you can edit, synchronize, or delete them.

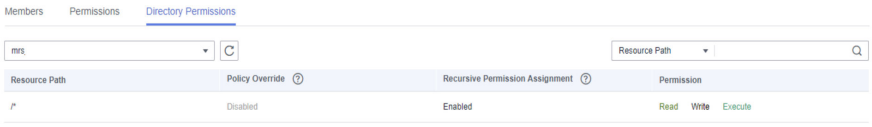
Figure 9-72 Configuring permissions on the By permissions page



Step 8 **Directory Permissions**: On the role details page, click the **Directory Permissions** tab.

Directory permissions obtain the HDFS policies of this role from the Ranger component to display the HDFS paths to which this role has permissions. In addition, you can view the operation permissions of the paths. You can search for the permissions of a path. Only exact match is supported.

Figure 9-73 Viewing directory permissions



Step 9 The permissions configured for the managed role do not take effect immediately. You need to manually synchronize the permissions to the Ranger component for permission management to take effect. For details, see **Synchronizing Permissions**.

----End

Related Operations

- Synchronizing permissions: After configuring data permissions on the **Role Management** page, you need to synchronize the permissions to the data source for permission management to take effect.

  To synchronize permissions, click **Synchronize Permissions** in the upper right corner of the **Basic Information** area on the role details page. To synchronize the permissions of multiple roles, select the roles in the role management navigation tree and click above the navigation tree.

- Synchronizing roles: In common role management (managed roles do not need to be synchronized), after a role is created for a permission set, the role takes effect only after being synchronized to the data source.

  To synchronize a role, click **Synchronize Role Information** in the upper right corner of the **Basic Information** area or click **Synchronize** in the **Operation** column on the **Data Source Role Configuration** tab page. To synchronize multiple roles, select the roles in the role management navigation tree and click above the navigation tree.

  NOTE

  - After role synchronization is successful, MRS data source roles are named in *Role name_Timestamp* format, and the GaussDB(DWS) data source roles are named in **dataarts_studio_role**_*Role name* format.
  - In scenarios where roles are synchronized to an MRS cluster, after the system prompts a successful role synchronization, permission management takes effect after about five minutes during which the Ranger component automatically synchronizes roles from the MRS cluster. You can check whether the synchronization is complete based on **Data Source Role Name** on the **Data Source Role Configuration** tab page.
    - Roles that are not synchronized are named in *Role name_10-digit timestamp* format.
    - Roles that have been synchronized are named in *Role name_13-digit timestamp* format.

- Deleting roles: In the **Role Management** navigation pane, select roles and click above the navigation pane. In the displayed dialog box, confirm the roles to be deleted and click **Yes**.

  Common roles for which roles, permissions, users, or child permission sets have been configured cannot be deleted. To delete such roles, delete the related configurations first. If permissions have been configured for a managed role, the role cannot be deleted. To delete the role, clear related configurations first.

  NOTE

  Deleted common roles are moved to the recycle bin. You can restore them within 30 days. After 30 days, they will be deleted permanently. For details, see **Managing the Recycle Bin**.

9.3.5.4 Managing Members

DataArts Security allows you to view the permissions of workspace members, and manage roles and permission sets.

Prerequisites

- Permission sets or roles have been configured for members. For details, see **Configuring Permission Sets** or **Configuring Roles**.

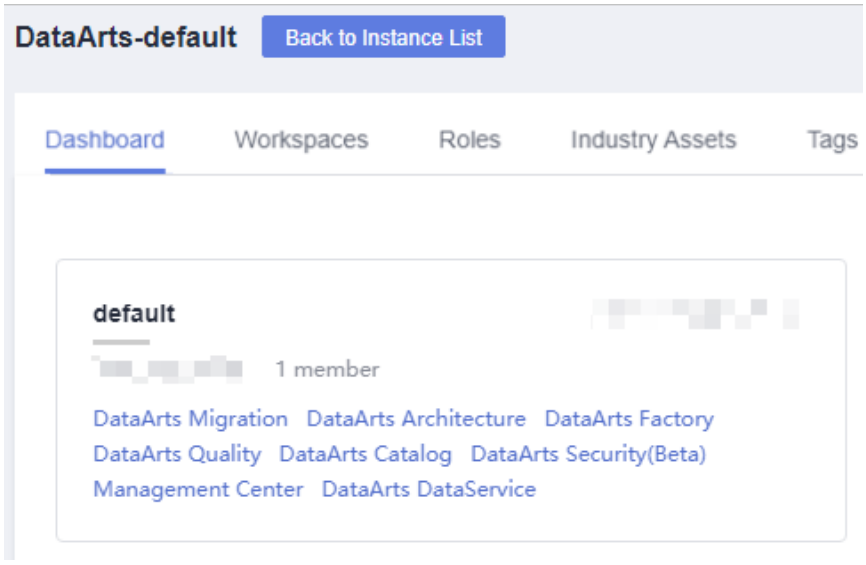
Constraints

- Only the DAYU Administrator, Tenant Administrator, data security administrator, or role or permission set administrator can add or delete roles or permission sets for members.
- Only common roles can be added or deleted for members. Managed roles are not supported.
- The permissions configured for members take effect only after roles or permission sets are successfully synchronized.

Viewing the Policy and Details

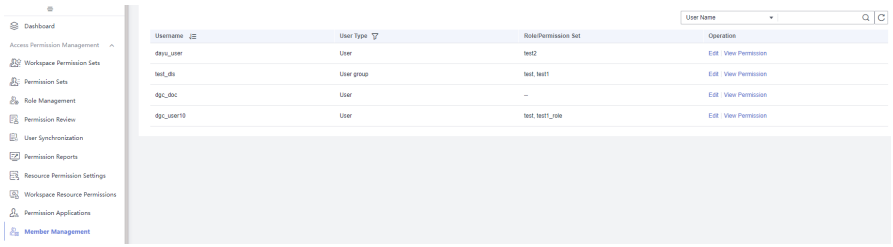
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-74 DataArts Security



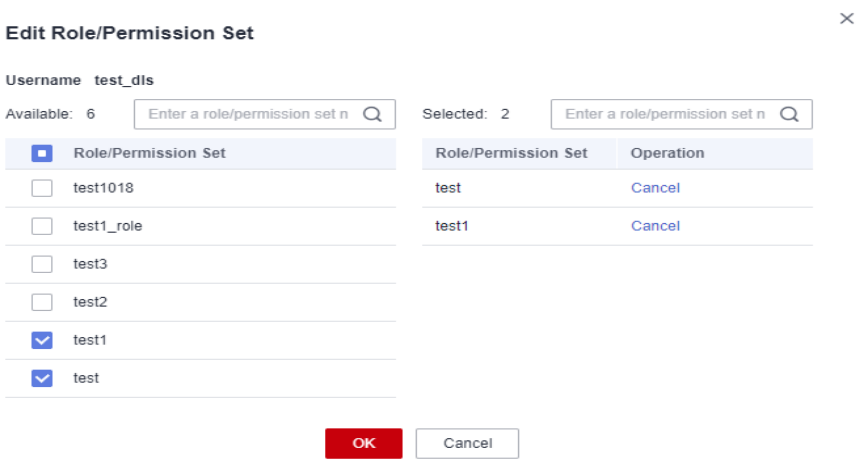
Step 2 In the navigation pane on the left, choose **Member Management**.

Figure 9-75 Member Management page



Step 3 Locate a member and click **Edit** in the **Operation** column. In the displayed dialog box, add or delete roles or permission sets for the member to manage its permissions.

Figure 9-76 Edit Role/Permission Set



Step 4 Click **View Permission** in the **Operation** column to view the basic information, permissions, and permission sources of a member.

----End

9.3.5.5 Configuring Row-level Access Control

Multiple developers may need to access and perform operations on the same GaussDB(DWS) table at the same time. In this case, you need to grant developers the permissions for specific rows in the table by configuring row-level access control policies.

After creating a row-level access control policy on the DataArts Security console, you can synchronize the policy to GaussDB(DWS). Row-level access control is automatically enabled for the GaussDB(DWS) table so that the policy takes effect.

The row-level access control policies configured for a DataArts Studio instance are visible to and take effect for all the workspaces of the instance.

Prerequisites

- Before creating a row-level access control policy, you have created a GaussDB(DWS) connection. For details, see **Creating a Data Connection**. The account in the GaussDB(DWS) connection must have the GRANT permission of the target table. (By default, only the owner of a database object or system administrator can run the **GRANT** command to grant the object permissions to other users.)
- Row-level access control policies need to be associated with data sources for specified users or user groups. Therefore, you need to synchronize user information from IAM to data sources first. For details, see **Synchronizing IAM Users to the Data Source**.
- If you want to use the current user identity authentication to make row-level access control policies take effect during script execution and job tests in DataArts Factory, you need to enable permission applications by following the instructions in **Enabling Fine-grained Authentication**.
- To ensure that a row-level access control policy takes effect, ensure that the user specified in the policy has the permission to the table to be controlled and has the USAGE permission of the schema to which the table belongs. You can run the following commands to grant permissions to user1, user2, and user3:

```
GRANT USAGE ON SCHEMA schema_name TO user1,user2,user3;
GRANT SELECT,UPDATE,DELETE ON TABLE table_name TO user1,user2,user3;
```

Constraints

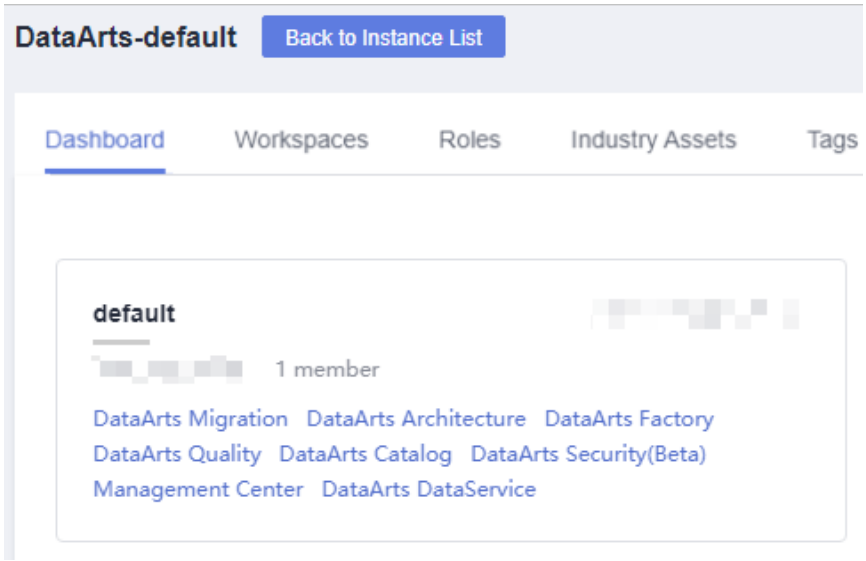
- Only the DAYU Administrator, Tenant Administrator, or data security administrator can create, modify, or delete row-level access control policies. Other common users do not have permission to perform these operations.
- Row-level access control policies are available for GaussDB(DWS) data sources and unavailable for GaussDB(DWS) logical clusters. The account in the GaussDB(DWS) connection must have the GRANT permission of the target table. (By default, only the owner of a database object or system administrator can run the **GRANT** command to grant the object permissions to other users.)

- Row-level access control policies need to be associated with data sources for specified users or user groups. Therefore, you need to synchronize user information from IAM to data sources first. For details, see **Synchronizing IAM Users to the Data Source**.
- Row-level access control supports read operations on data tables (SELECT, UPDATE, DELETE, and ALL), and does not support write operations on data tables (INSERT and MERGE INTO).
- A row-level access control policy name is specific to a table. A data table cannot have row-level access control policies with the same name. Different data tables can have the same row-level access control policy.
- Row-level access control policies can be defined for row-store tables, row-store partitioned tables, column-store tables, column-store partitioned tables, replication tables, unlogged tables, and hash tables. Row-level access control policies cannot be defined for HDFS tables, foreign tables, or temporary tables.
- Row-level access control policies cannot be defined for views.
- A maximum of 100 row-level access control policies can be defined for a table.
- Users with GaussDB(DWS) administrator permissions and the initial O&M user (Ruby) are not affected by row-level access control. They can view all the data of a table.
- Tables queried by using SQL statements, views, functions, and stored procedures are affected by row-level access control policies.
- After a row-level access control policy is synchronized, the types of the columns on which the row-level access control policy depends cannot be changed.

Create a Row-Level Access Control Policy

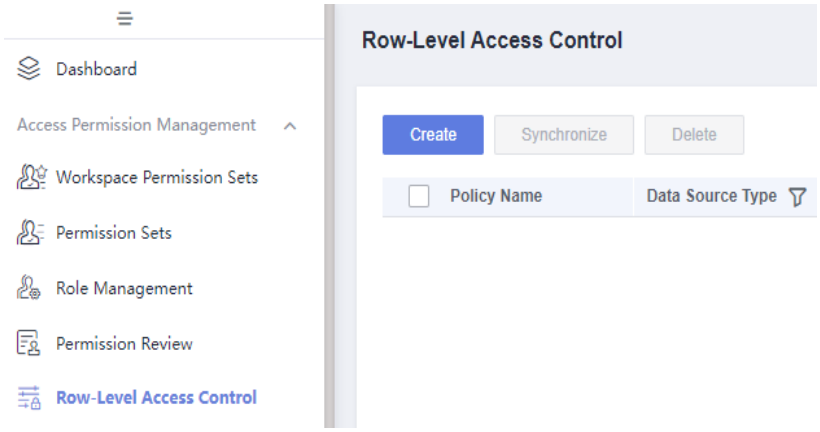
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-77 DataArts Security



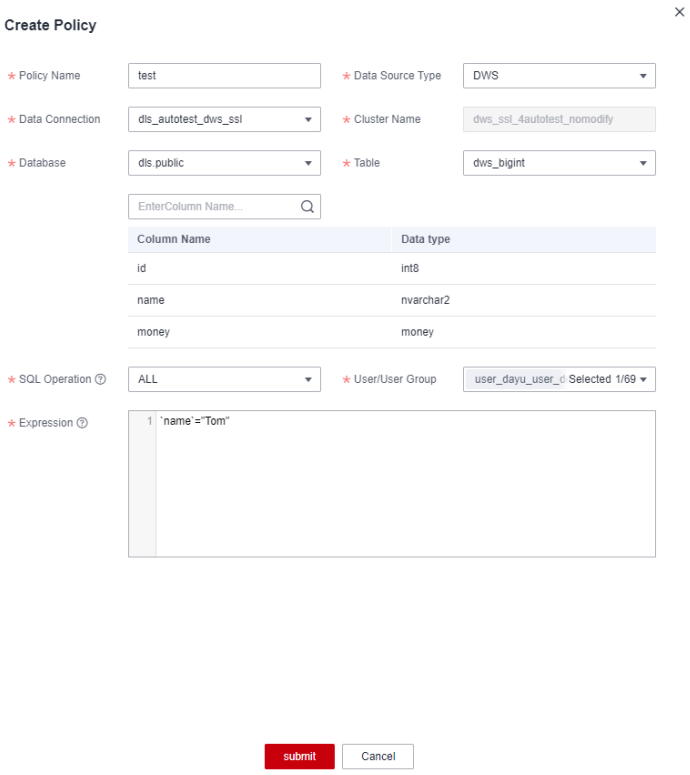
Step 2 In the navigation pane on the left, choose **Row-Level Access Control**.

Figure 9-78 Row-Level Access Control page



Step 3 Click **Create** and set the parameters listed in **Table 9-7**.

Figure 9-79 Setting the parameters for creating a row-level access control policy



The following table lists the parameters for creating a row-level access control policy.

Table 9-7 Policy parameters

Parameter	Description
*Policy Name	Name of the row-level access control policy. It must be unique for a data table. To facilitate policy management, you are advised to include the target object and content rule in the name.
*Data Source Type	Only **DWS** is supported.
*Data Connection	If no data connection is available, create one by referring to **Creating a Data Connection**.
*Cluster Name	You do not need to set this parameter. The data source cluster in the data connection is automatically selected.
*Database	Database where the row is located
*Table	Table where the row is located. After you select a table, the table structure is automatically displayed.
*SQL Operation	Select the operation to be controlled (**SELECT**, **UPDATE**, **DELETE**, or **ALL**). Write operations including INSERT and MERGE INTO are not supported. • If you select **SELECT**, SELECT operations will be controlled by the policy. The selected user group or user can only view the rows that meet the conditions defined by the expression. The affected operations include SELECT, UPDATE ... RETURNING, and UPDATE ... RETURNING. • If you select **UPDATE**, UPDATE operations will be controlled by the policy. The selected user group or user can only update the rows that meet the conditions defined by the expression. The affected operations include UPDATE, UPDATE ... RETURNING, and SELECT ... FOR UPDATE/SHARE. • If you select **DELETE**, DELETE operations will be controlled by the policy. The selected user group or user can only delete the rows that meet the conditions defined by the expression. The affected operations include DELETE and DELETE ... RETURNING.

Parameter	Description
*User Group/ User	Select the user or user group from the current workspace members. The specified user or user group can perform the selected SQL operation only on the row-level data that meets the condition defined by the expression. • If you select **SELECT**, SELECT operations will be controlled by the policy. The selected user group or user can only view the rows that meet the conditions defined by the expression. The affected operations include SELECT, UPDATE ... RETURNING, and UPDATE ... RETURNING. • If you select **UPDATE**, UPDATE operations will be controlled by the policy. The selected user group or user can only update the rows that meet the conditions defined by the expression. The affected operations include UPDATE, UPDATE ... RETURNING, and SELECT ... FOR UPDATE/ SHARE. • If you select **DELETE**, DELETE operations will be controlled by the policy. The selected user group or user can only delete the rows that meet the conditions defined by the expression. The affected operations include DELETE and DELETE ... RETURNING.

*Expression	Enter the expression for determining the row data. The specified user or user group can perform the selected SQL operation only on the rows of data that meet the condition defined by the expression. The expression is in the following format:
	`` `*Target field*`="*Operation value*" ``
	You are advised to enclose target fields in backquotes and enclose operation values in double quotation marks. Use **AND** to combine multiple rows of data to be matched. The following is an example.
	``` `role`="test" AND `department`="sales" ```

Step 4 Click **Submit**. After the row-level access control policy is created, click **Synchronize** to synchronize the policy to the data source.

----End

Related Operations

- Synchronizing a policy: On the **Row-Level Access Control** page, locate a policy and click **Synchronize** in the **Operation** column to synchronize the policy to the data source. To synchronize multiple policies, select them and click **Synchronize** above the list.

  Policies take effect only after they are synchronized successfully. If the policy synchronization fails, you can view the policy run log in the **policy details** to locate the failure cause. After rectifying the fault, synchronize the policy again. If the synchronization still fails, contact technical support.

- Editing a policy: On the **Row-Level Access Control** page, locate a policy and click **Edit** in the **Operation** column.
- Deleting policies: On the **Row-Level Access Control** page, locate a policy and click **Delete** in the **Operation** column. To delete multiple policies, select them and click **Delete** above the list.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.
- Viewing policy details: On the **Row-Level Access Control** page, locate a policy and click its name to view its details.

  Figure 9-80 Viewing policy details

9.3.5.6 Synchronizing MRS Hive and Hetu Permissions

If MRS Hetu is connected to MRS Hive and Ranger is used for permission control, the Ranger permissions of Hetu rather than of Hive are used to authenticate the access to Hive data from Hetu in the same cluster.

To avoid repeated configuration of Hive data permissions on Hetu, you can configure a Hetu permission synchronization policy so that Hive permissions can be automatically synchronized to Hetu. This improves permission management consistency and usability.

The Hetu permission synchronization policies configured for a DataArts Studio instance are visible to and take effect for all the workspaces of the instance.

Prerequisites

- Ranger permission control has been enabled for MRS Hetu. For details, see **HetuEngine Permission Management Overview**.
- Before configuring a Hetu permission synchronization policy, you have created an MRS Hive connection and an MRS Hetu connection in Management Center. For details, see **Creating a Data Connection**.

Constraints

- Only the DAYU Administrator, Tenant Administrator, or data security administrator can create, modify, or delete Hetu permission synchronization

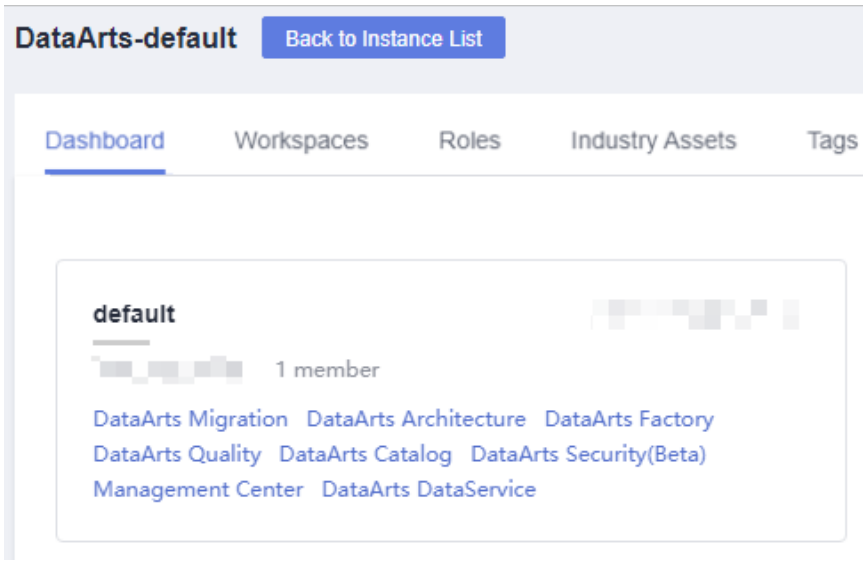
policies. Other common users do not have permission to perform these operations.

- Hive permissions can be synchronized only to Hetu in the same MRS cluster.
- When configuring a Hetu permission synchronization policy, you need to configure mappings between Hive and Hetu catalogs. If a Hive source is connected to multiple Hetu catalogs, you need to configure multiple synchronization policies.
- After a Hetu permission synchronization policy is created, existing Hive permissions will not be automatically synchronized to Hetu. Instead, the permissions will be synchronized to Hetu only after a permission synchronization is triggered. This prolongs the permission synchronization duration.
- Hive permission synchronization is not affected if permissions fail to be synchronized to Hetu.
- After a Hetu permission synchronization policy is deleted, the permissions that have been synchronized to Hetu will not be revoked.
- The names of Ranger policies for synchronizing permissions to Hetu are in the following format: ***Catalog name_Schema name*+*Table name*+*Column name***. If a policy with the same resource and name already exists on Hetu Ranger, permissions will fail to be synchronized to Hetu. In this case, you must manually clear that existing policy on Hetu Ranger.

Creating a Hetu Permission Synchronization Policy

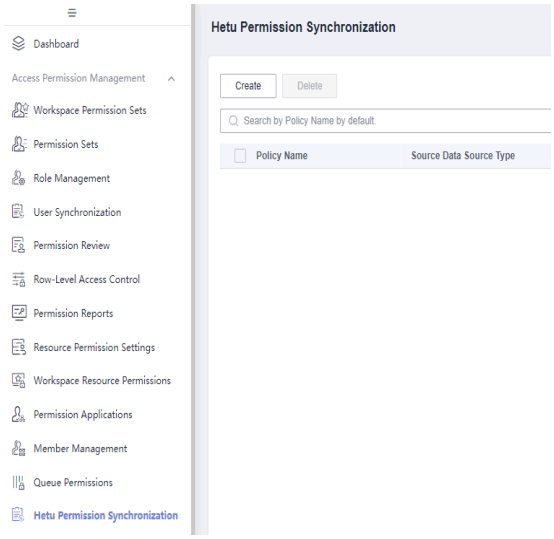
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-81 DataArts Security



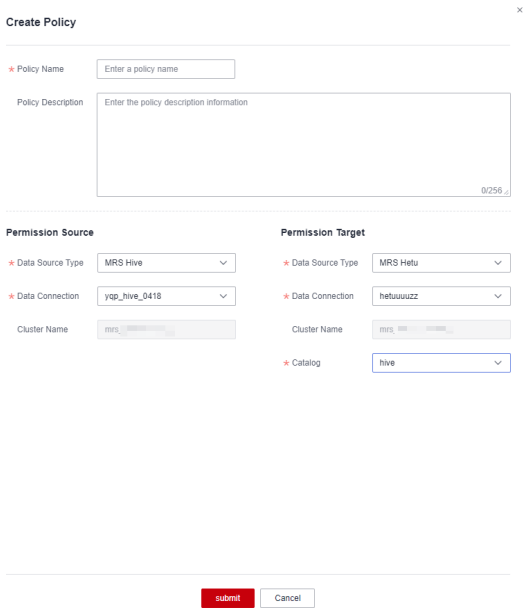
Step 2 In the left navigation pane, choose **Hetu Permission Synchronization**.

Figure 9-82 Hetu Permission Synchronization page



Step 3 Click **Create** and set the parameters listed in **Table 9-8**.

Figure 9-83 Setting parameters for a Hetu permission synchronization policy



The following table lists the parameters for a Hetu permission synchronization policy.

Table 9-8 Policy parameters

Parameter	Description
*Policy Name	Name of the Hetu permission synchronization policy. It must be unique for each data table. You are advised to include the cluster name and catalog name in the policy name for easy management.
Policy Description	A description of the Hetu permission synchronization policy to be created. It can contain a maximum of 255 characters.
Permission Source	
*Data Source Type	Only **MRS Hive** is supported.
*Data Connection	If no data connection is available, create one by referring to **Creating a Data Connection**.
Cluster Name	The data source cluster in the data connection is automatically selected.
Permission Target	
*Data Source Type	Only **MRS Hetu** is supported.
*Data Connection	If no data connection is available, create one by referring to **Creating a Data Connection**. The cluster to which the selected Hetu connection belongs must be the same as that to which the Hive connection belongs.
Cluster Name	The data source cluster in the data connection is automatically selected.
*Catalog	Name of the Hetu data source, which is **hive** by default. Multiple Hetu catalogs can connect to the same Hive. You can also select another catalog of the cluster.

Step 4 Click **Submit**.

Step 5 When Hive permission synchronization is triggered, permissions are synchronized to Ranger on Hetu. The policy is named in the following format: ***Catalog name_Schema name+Table name+Column name***. **Table 9-9** shows the policy mapping between Hive and Hetu.

Table 9-9 Policy mapping between Hive and Hetu

Hive	Hetu
Resource mapping	
Hive data source	Hetu Catalog

Hive	Hetu
Hive database	Hetu Schema
Hive table	Hetu table
Hive column	Hetu column
Permission mapping	
select	select and use
update	insert, delete, and update
create	create
drop	drop
alter	alter
all	all

----End

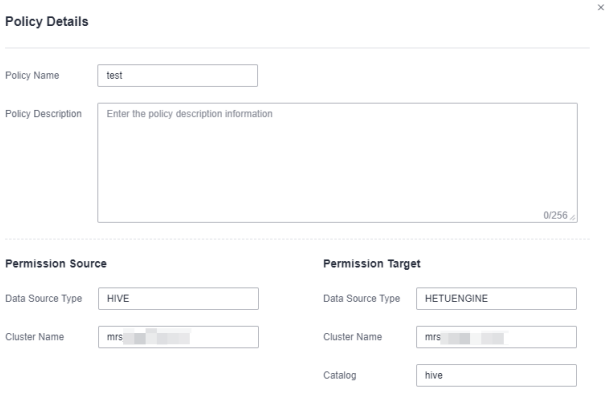
Related Operations

- Editing a policy: On the **Hetu Permission Synchronization** page, locate a policy and click **Edit** in the **Operation** column to edit the policy.
- Deleting policies: On the **Hetu Permission Synchronization** page, locate a policy and click **Delete** in the **Operation** column to delete the policy. To delete multiple policies, select them and click **Delete** above the policy list.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.
- Viewing policy details: On the **Hetu Permission Synchronization** page, locate a policy, and click **Details** in the **Operation** column to view details of the policy.

  Figure 9-84 Viewing policy details



9.3.5.7 Applying for Permissions and Reviewing Permission Requests

During access permission management, you can grant permissions to users through permission sets or roles, or apply for permissions and approve permission applications.

This section describes how an applicant applies for permissions (**Applying for Permissions**) and how a reviewer reviews permission requests (**Reviewing Permission Requests**) and revokes permissions (**Revoking Permissions**).

Prerequisites

- Before applying for permissions, you have configured a workspace permission set. For details, see **Configuring Workspace Permission Sets**.
- Before applying for permissions, you have collected the metadata of the data connection in DataArts Catalog. For details, see **Metadata Collection Task**.

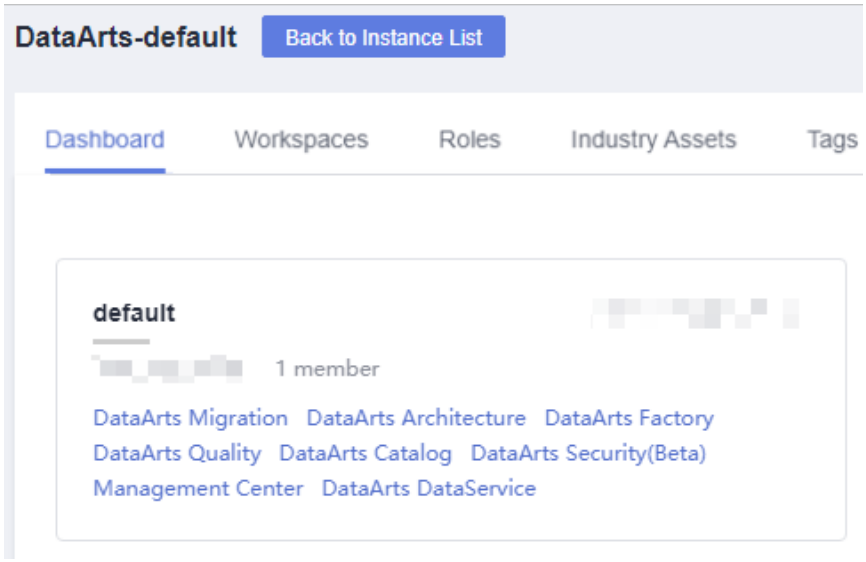
Constraints

- You can only apply for the SELECT permission for querying data in tables. Before applying for the permission, ensure that the SELECT permission for all columns in the target table has been configured in the workspace permission set.
- Only the DAYU Administrator, Tenant Administrator, data security administrator, and workspace administrator can revoke permissions from other users.
- If you apply for the permission of multiple tables at a time, multiple requests are generated.
- You can only view your permission requests and approval records, and cannot audit permissions.
- You can apply for DLI permissions for users but not for user groups.
- Only the DAYU Administrator, Tenant Administrator, workspace administrator, and data security administrator can revoke data permissions of users in the corresponding workspace.
- During permission synchronization, you need to configure required permissions for the **dlg_agency**. For details, see **Authorizing dlg_agency**.
- The current data permission control uses the allowlist mechanism, which adds operation conditions to the users to be authorized without affecting the permissions the users already have. If you only want to make the permissions granted by the data permission control take effect, you need to revoke the original permissions of the users to be authorized. For details, see **Data Permission Management**.
- During script execution and job testing in DataArts Factory, the MRS or GaussDB(DWS) data source uses the account of the data connection for authentication by default. Therefore, permission management still does not take effect during data development. You need to enable fine-grained authentication so that the current user is used for authentication during script execution and job testing in DataArts Factory. In this way, different users have different data permissions, and permission management for roles and permission sets takes effect.

Applying for Permissions

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

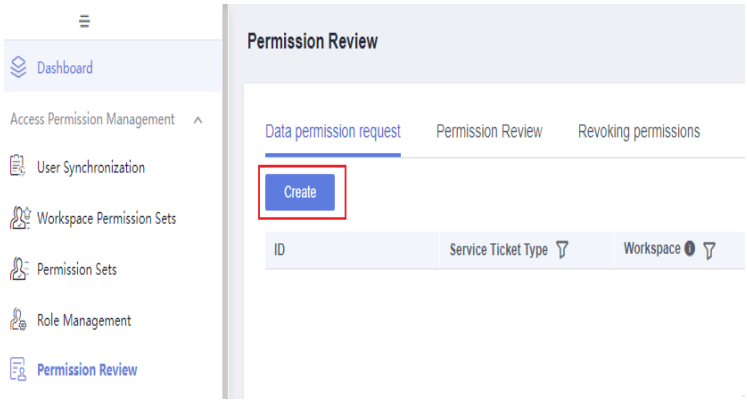
Figure 9-85 DataArts Security



Step 2 In the left navigation pane, choose **Permission Review**.

Step 3 On the **Data permission request** page, click **Create** to create a service ticket for applying for permissions.

Figure 9-86 Creating a permission request



Step 4 On the displayed **Data permission request** page, fill in the service ticket by referring to **Table 9-10**.

Figure 9-87 Filling in the service ticket

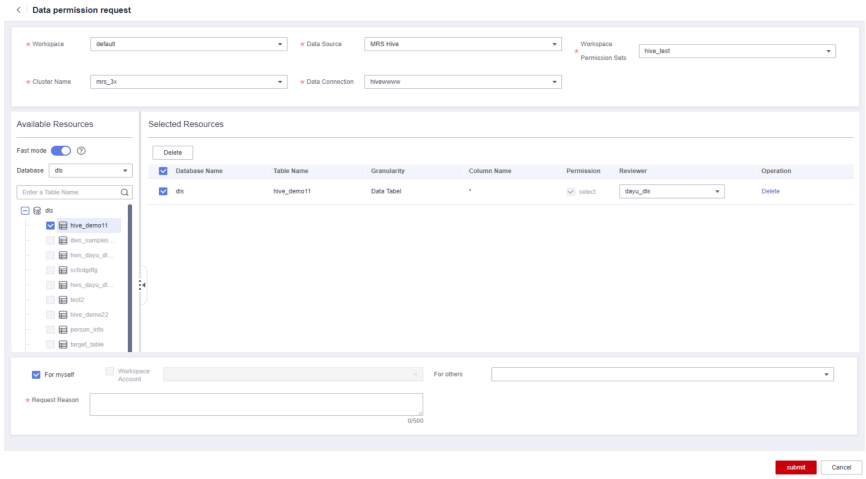


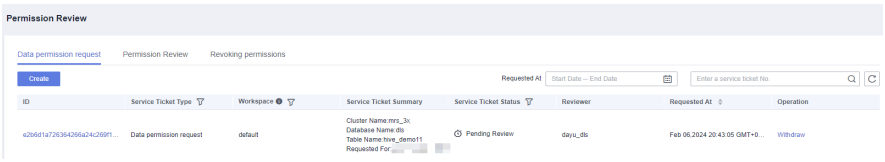
Table 9-10 Parameters for the permission request

Item	Description
Basic information	
*Workspace	Select a workspace for which a workspace permission set has been configured.
*Workspace Permission Sets	Select a workspace permission set that contains the required resource permissions.
*Data Source	Select **MRS Hive**, **DLI**, or **DWS**.
*Cluster Name	Select the cluster of the requested resource permissions.
*Data Connection	Select the data connection of the requested resource permissions.
Resource selection	
*Available Resources	After selecting a database in the navigation tree, select the required data tables. You can select tables in different databases. **NOTE** You can only apply for the SELECT permission for querying data in tables. Before applying for the permission, ensure that the SELECT permission for all columns in the selected table has been configured in the workspace permission set. If **Fast mode** is enabled, metadata of databases, tables, and columns is obtained from DataArts Catalog. Otherwise, metadata is obtained from the data source. You are advised to enable **Fast mode**.

Item	Description
*Selected Resources	In the list of selected resources, you can view the selected tables, permissions, and reviewers. **NOTE** The reviewers are the administrators of permission sets or roles. For example, if the SELECT permission for all columns in the selected table is defined in the workspace permission set, permission set A, and role B, the reviewer can be the administrator of permission set A or role B. If the SELECT permission for all columns in the selected table is only defined in the workspace permission set, the reviewer is the administrator of the workspace permission set.
Request information	
For myself	If you select this option, you can apply for the selected resource permissions for yourself.
Workspace Account	If a public IAM account for scheduling has been configured in DataArts Factory, you can apply for the selected resource permissions for the workspace account.
For others	Select members in the workspace and apply for the selected resource permissions for them.
*Request Reason	Enter the reason for applying for the permissions so that the reviewer can determine whether to approve your request.

Step 5 After filling in the service ticket, click **Submit** to generate a service ticket to be reviewed. In the service ticket list, you can view the service ticket ID, summary, and status. You can click the service ticket ID to view the ticket details. You can also withdraw service tickets that have not been approved.

Figure 9-88 Service ticket list

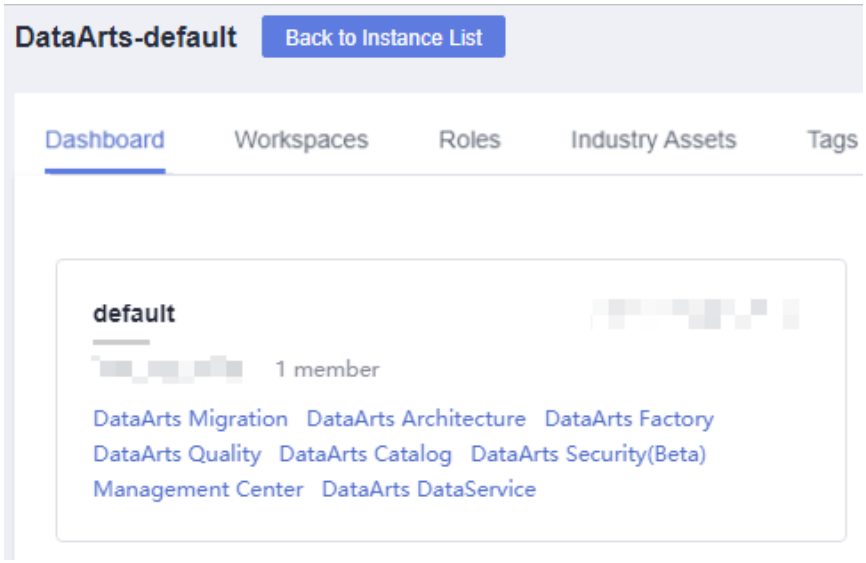


----End

Reviewing Permission Requests

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

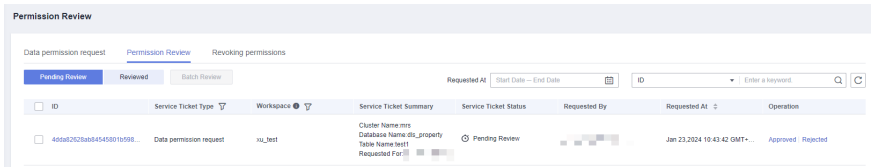
Figure 9-89 DataArts Security



Step 2 In the left navigation pane, choose **Permission Review**.

Step 3 Click the **Permission Review** tab.

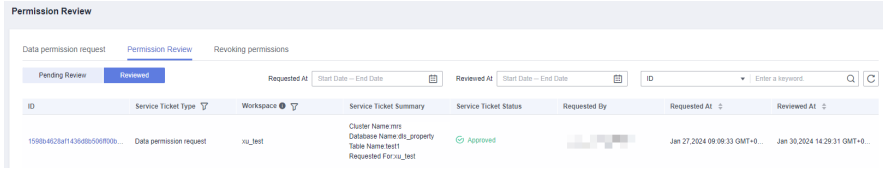
Figure 9-90 Permission Review



Step 4 The **Permission Review** page displays the service tickets to be reviewed. You can view the service ticket ID, summary, and status, and click the service ticket ID to view the ticket details. Review the service ticket based on service rationality and data security, and click **Approve** or **Reject**. You can also select service tickets and click **Batch Review** above the list to approve or reject service tickets in batches.

Step 5 Click the **Reviewed** tab to view the service tickets that have been approved.

Figure 9-91 List of approved service tickets

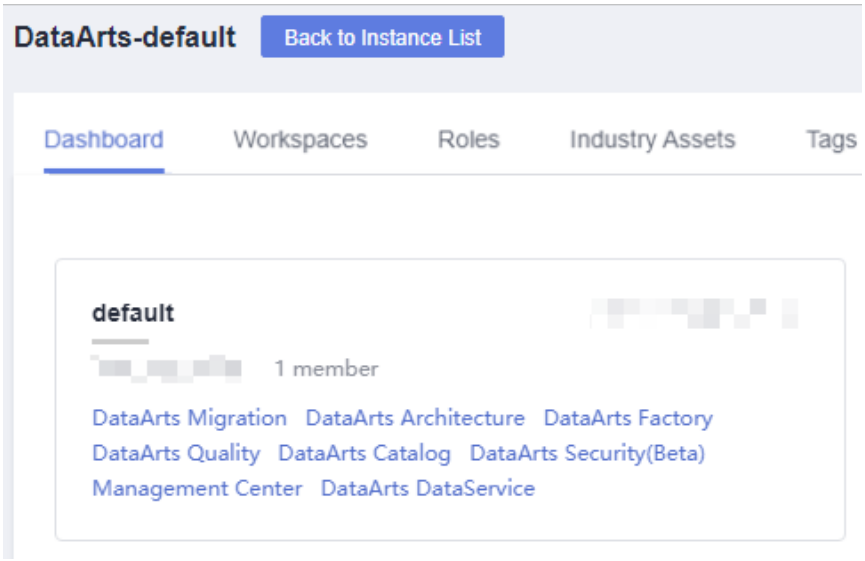


----End

Revoking Permissions

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

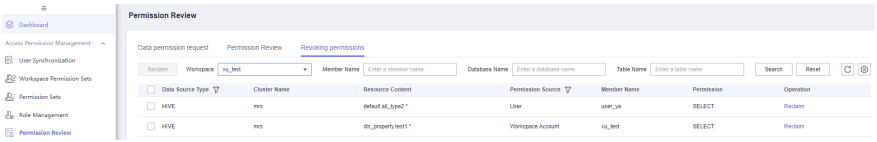
Figure 9-92 DataArts Security



Step 2 In the left navigation pane, choose **Permission Review**.

Step 3 Click the **Revoking permissions** tab.

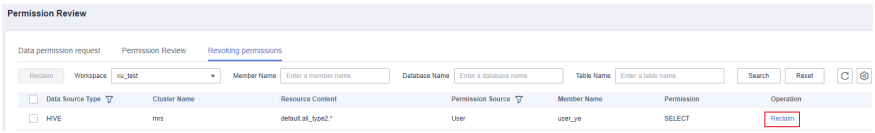
Figure 9-93 Revoking permissions tab



Step 4 The **Revoking permissions** page displays the data permissions you have obtained. You can filter permissions by **Workspace**, **Member Name**, **Database Name**, or **Table Name** (fuzzy match is supported). Locate a permission and click **Reclaim** in the **Operation** column to delete the permission.

Only the DAYU Administrator, Tenant Administrator, workspace administrator, and data security administrator can revoke data permissions of users in the corresponding workspace.

Figure 9-94 Revoking permissions



----End

9.3.5.8 Enabling Fine-grained Authentication

During script execution and job testing in DataArts Factory, the data source uses the account of the data connection for authentication. Therefore, user permission management enabled through roles or permission sets does not take effect for data development.

After fine-grained authentication is enabled, data sources no longer use the accounts of the data connections during script execution, job tests, and job scheduling in DataArts Factory of DataArts Studio. Instead, the current user is used for authentication. In this way, different users have different data permissions, and the permissions of roles and permission sets can be managed.

The impact of fine-grained authentication on the execution of scripts and jobs in DataArts Factory is as follows:

- If fine-grained authentication is disabled, the account of the data connection is used for authentication during script execution and job tests and scheduling in DataArts Factory.
- If fine-grained authentication is enabled for development mode, the current user is used for authentication during script execution and job tests in DataArts Factory, and the account of the data connection is used for authentication during scheduling.
- If fine-grained authentication is enabled for scheduling mode, the current user is used for authentication during script execution, job tests, and job scheduling in DataArts Factory.

Prerequisites

- Proxy permissions have been configured for the users of an MRS Hive connection and MRS Spark connection. For details, see **Reference: Configuring Proxy Permissions for MRS Data Connection Users**.
- The Spark2x component corresponding to the MRS Spark connection uses the multi-active instance mode. Otherwise, change the mode to the multi-active instance mode by referring to **Configuring the Switchover Between the Multi-active Instance Mode and the Multi-tenant Mode**.
- Required data permissions have been configured for the user who uses the data source to prevent service interruptions due to insufficient data permissions after fine-grained authentication is enabled. For details about how to configure permissions, see **Configuring Permission Sets** or **Configuring Roles**.
- Before testing the connectivity of a DWS connection, you have switched the current login account to an IAM sub-user.

Constraints

- Fine-grained authentication for development mode is available only for GaussDB(DWS) and MRS Hive and MRS Spark data sources in proxy mode. Fine-grained authentication for scheduling mode is available only for the MRS Hive data source in proxy mode.
- Only the DAYU Administrator, Tenant Administrator, and data security administrator have the permission to configure fine-grained authentication.
- During the connectivity test, the system uses the current user account to access the data source to ensure that the current user can access the data

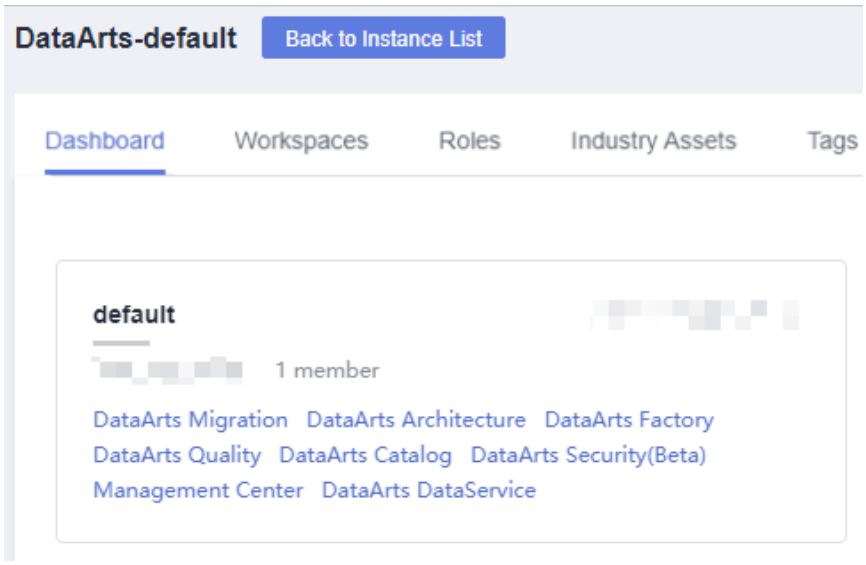
source. DWS data sources cannot be accessed using a a Huawei account. Therefore, if you log in using a a Huawei account, the connectivity test will fail. Before testing the connectivity of a DWS connection, you need to switch the current login account to an IAM sub-user.

- Fine-grained authentication is supported only when the version of the CDM cluster selected for the agent in the data connection is 2.10.0.300 or later.
- Fine-grained authentication is supported only when the guest_agent version of a GaussDB(DWS) cluster is 8.2.1, or later than 8.2.1 and earlier than 9.0.0. For details about how to check the guest_agent version of a GaussDB(DWS) cluster, see **Viewing the guest_agent Version of a GaussDB(DWS) Cluster**.
- Fine-grained authentication is supported only when proxy permissions are configured for the users of MRS Hive and MRS Spark data connections.
- Fine-grained authentication is supported only when the Spark 2x component corresponding to the MRS Spark data connection uses the multi-active instance mode. For details about how to change the mode to the multi-active instance mode, see **Configuring the Switchover Between the Multi-active Instance Mode and the Multi-tenant Mode**.
- User permissions configured in a role/permission set take effect only after the role/permission set is successfully synchronized and fine-grained authentication is enabled.

Enabling Fine-grained Authentication

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-95 DataArts Security



Step 2 In the left navigation pane, choose **Permissions Applications**.

Step 3 On the **Permissions Applications** page, test the connectivity of the data connections for which you want to enable fine-grained authentication. During the connectivity test, the system uses the current user account to access the data source to ensure that the current user can access the data source.

> **NOTE**
>
> DWS data sources cannot be accessed using a a Huawei account. Therefore, if you log in using a a Huawei account, the connectivity test will fail. Before testing the connectivity of a DWS connection, you need to switch the current login account to an IAM sub-user.

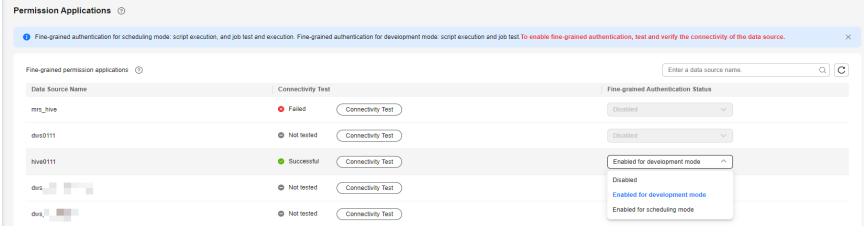
Figure 9-96 Testing connectivity

If the connectivity is abnormal, perform the following checks:

1. Check whether the data source connected by the data connection is available.
2. Check whether the version of the CDM cluster selected as the agent for the data connection is 2.10.0.300 or later.
3. Check whether the guest_agent version of the GaussDB(DWS) cluster connected by the data connection is 8.2.1 or later than 8.2.1 and earlier than 9.0.0. For details about how to check the guest_agent version of a GaussDB(DWS) cluster, see **Viewing the guest_agent Version of a GaussDB(DWS) Cluster**.
4. Check whether a proxy has been configured for the user of the MRS Hive or MRS Spark connection. If no proxy has been configured, see **Reference: Configuring Proxy Permissions for MRS Data Connection Users**.
5. Check whether the Spark 2x component corresponding to the MRS Spark data connection uses the multi-active instance mode. Fine-grained authentication is supported only in multi-active instance mode. For details about how to change the mode to the multi-active instance mode, see **Configuring the Switchover Between the Multi-active Instance Mode and the Multi-tenant Mode**.

Step 4 After the connectivity test is successful, select **Enabled for development mode** or **Enabled for scheduling mode** in the **Fine-grained Authentication Status** column, and click **Submit** to enable fine-grained authentication.

Figure 9-97 Enabling fine-grained authentication



----End

Reference: Configuring Proxy Permissions for MRS Data Connection Users

By default, when you access a data source through an MRS Hive or Spark data connection on DataArts Studio, the account configured in the data connection is

used by default. If you configure Hive or Spark proxy permissions for the account in the MRS Hive or Spark data connection, you can use your own identity to perform this operation and enable fine-grained authentication. For details, see **Configuring Hive proxy permissions** and **Configuring Spark proxy permissions**.

Configuring Hive proxy permissions

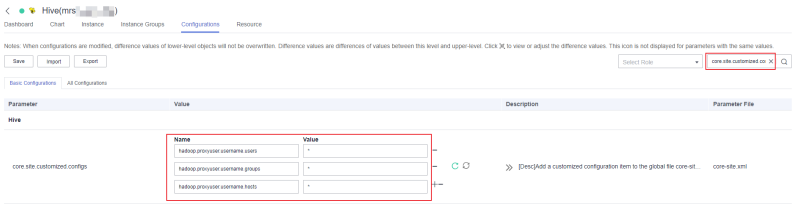
Step 1 Log in to MRS FusionInsight Manager.

Step 2 Choose **Cluster** > **Services** > **Hive** and click **Configurations** and then **Basic Configurations**. In the search box, enter **core.site.customized.configs** and set the parameter listed in **Figure 9-98**.

Table 9-11 Parameter

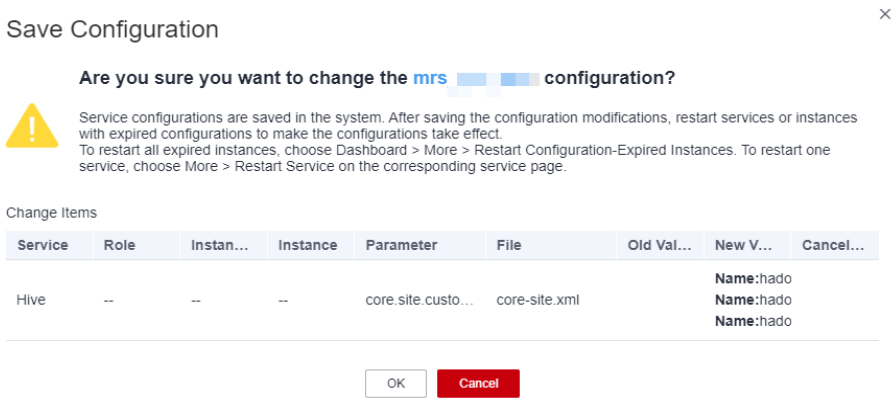
Parameter	Name	Value
core.site.customized.configs	hadoop.proxyuser.***Username configured for the data connection***.users	*
	hadoop.proxyuser.***Username configured for the data connection***.groups	*
	hadoop.proxyuser.***Username configured for the data connection***.hosts	*

Figure 9-98 Configuring the core.site.customized.configs parameter



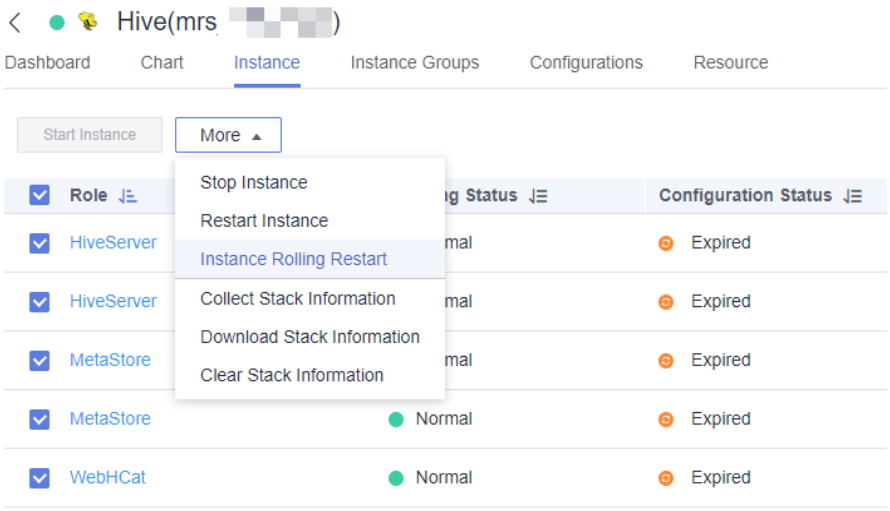
Step 3 After setting the parameter, click **Save** in the upper left corner and then **OK** in the dialog box to save the configuration.

Figure 9-99 Saving the configuration



Step 4 After saving the configuration, switch to the **Instances** tab page, select the instance that has expired, click **More**, and select **Instance Rolling Restart** to make the configuration take effect.

Figure 9-100 Performing a rolling instance restart



----End

Configuring Spark proxy permissions

Step 1 Log in to MRS FusionInsight Manager.

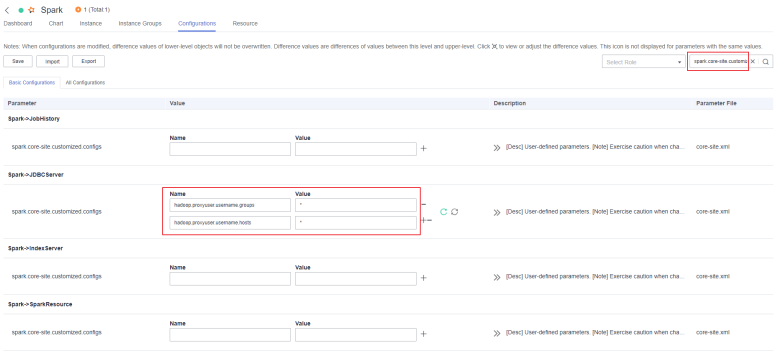
Step 2 Choose **Cluster** > **Services** > **Spark** and click **Configurations** and then **Basic Configurations** or choose **Cluster** > **Services** > **Spark2x** and click **Configurations** and then **Basic Configurations**. In the search box, enter **spark.core-site.customized.configs** and set the parameter listed in **Figure 9-101**. The Spark component is used as an example.

Table 9-12 Parameter

Parameter		Name	Value
Spark->JDBCServer Or Spark2x->JDBCServer2x	core.site.customized.configs	hadoop.proxyuser.***Username configured for the data connection***.groups	*
		hadoop.proxyuser.***Username configured for the data connection***.hosts	*
		hadoop.proxyuser.***Username configured for the data connection***.groups	*

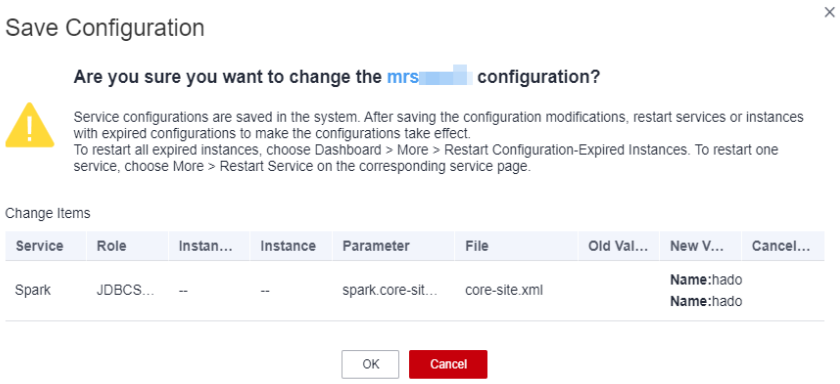
Parameter		Name	Value
		hadoop.proxyuser.***Username configured for the data connection***.hosts	*

Figure 9-101 Configuring the spark.core-site.customized.configs parameter



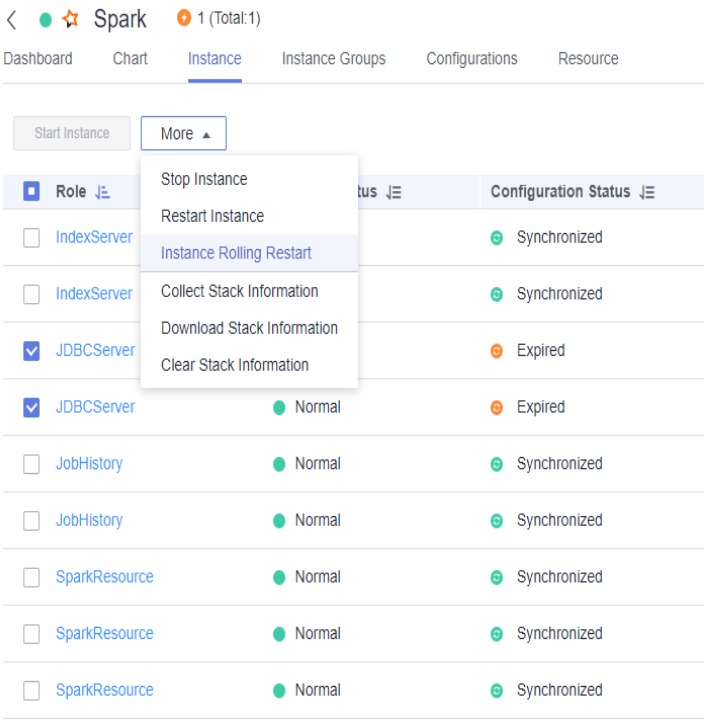
Step 3 After setting the parameter, click **Save** in the upper left corner and then **OK** in the dialog box to save the configuration.

Figure 9-102 Saving the configuration



Step 4 After saving the configuration, switch to the **Instances** tab page, select the instance that has expired, click **More**, and select **Instance Rolling Restart** to make the configuration take effect.

Figure 9-103 Performing a rolling instance restart



----End

9.3.6 Controlling Service Resource Access

9.3.6.1 Configuring Queue Permissions

This section describes how to allocate MRS Yarn and DLI queues to the current workspace and configure queue permission policies for user groups or users through queue permission management.

Currently, the whitelist mechanism is used for queue allocation and queue permission management. If no queue is allocated, no queue can be selected. If queue permissions are not granted to a user, the user cannot use the queue.

- After queues are allocated to the workspace, they can be selected during the job node configuration in DataArts Factory.

  NOTE

  Currently, the queue list can be obtained from allocated queues when the MRS Yarn queue is selected. If no queue is allocated, only the root.default queue can be selected.

- After queue permissions are configured for user groups or users, MRS Ranger manages the permissions of MRS queues and DLI manages the permissions of DLI queues. Only authorized users can access the queues.

NOTE

When you use queues in DataArts Factory, the data source uses the account of the data connection for authentication. Therefore, queue permission management still does not take effect during data development. You need to enable fine-grained authentication so that the current user is used for authentication during the use of queues in DataArts Factory. In this way, queue permission management takes effect.

Prerequisites

- Only the DAYU Administrator, Tenant Administrator, or data security administrator has the permission to allocate available queues to the current workspace, configure MRS queue attributes (offline/real-time), and configure user permission policies for specified queues. The workspace administrator can configure queue permission policies for user groups and users.
- Before configuring queue permissions, you have created an MRS Ranger and a DLI connection in Management Center. For details, see **Creating a Data Connection**.
- Before configuring permissions for MRS Yarn queues, you have synchronized user information from IAM to the data source based on **Synchronizing IAM Users to the Data Source**.
- To make the permission policy for MRS Yarn queues take effect, you have enabled Yarn access control by setting the **yarn.acl.enable** parameter to **true**. For details, see **Reference: Configuring Strict Permission Control for Yarn**.

Constraints

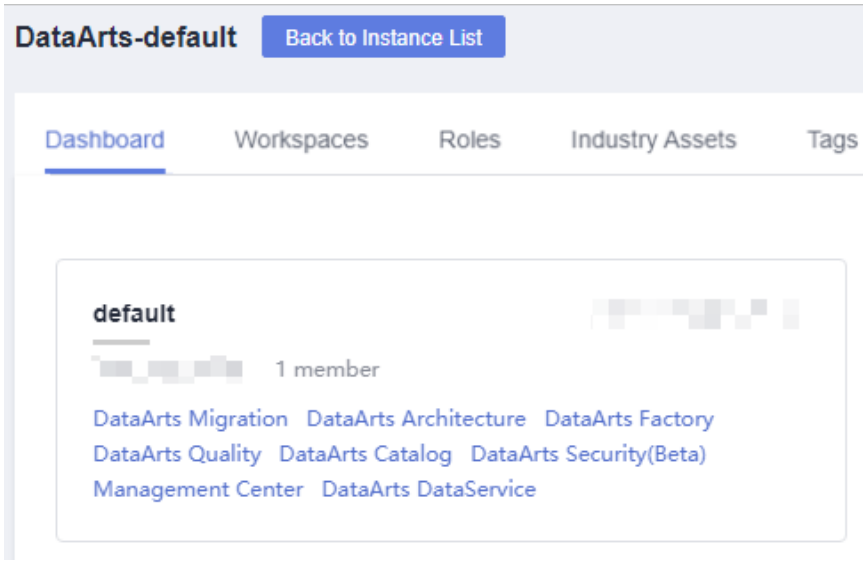
- Currently, only MRS Yarn queues can be allocated. Permission management is supported only for MRS Yarn and DLI queues. Authorization for the DLI default queue is not supported due to DLI limitations.
- Permissions of MRS Yarn queues can be managed only when the version of the CDM cluster selected as the agent for the data connection is 2.10.0.300 or later.
- Only the DAYU Administrator, Tenant Administrator, or data security administrator has the permission to allocate available queues to the current workspace, configure MRS queue attributes (offline/real-time), and configure user permission policies for specified queues. The workspace administrator can configure queue permission policies for user groups and users.
- The queues allocated to the current workspace are not associated with the configured queue permissions policies which are contained in the data source configuration. Therefore, if the queues are deleted from the current workspace, the configured queue permission policies still take effect. When the queues are added again, the permissions are still available.
- The configured queue permission policies are implemented based on the permission control capability of the data source. You can view the configured policies in the data source (such as MRS Ranger policies and DLI queue management). If you delete a queue policy from the data source, the policy will not be automatically deleted from the DataArts Security component. You need to manually delete the policy from the DataArts Security component.
- Queue attributes (offline or real-time) can be configured only for MRS Yarn queues, and different attributes can be configured for the same queue in different workspaces.

- Due to DLI limitations, permissions of DLI queues can be granted only to users, but not to user groups.

Allocating Queues and Granting Permissions

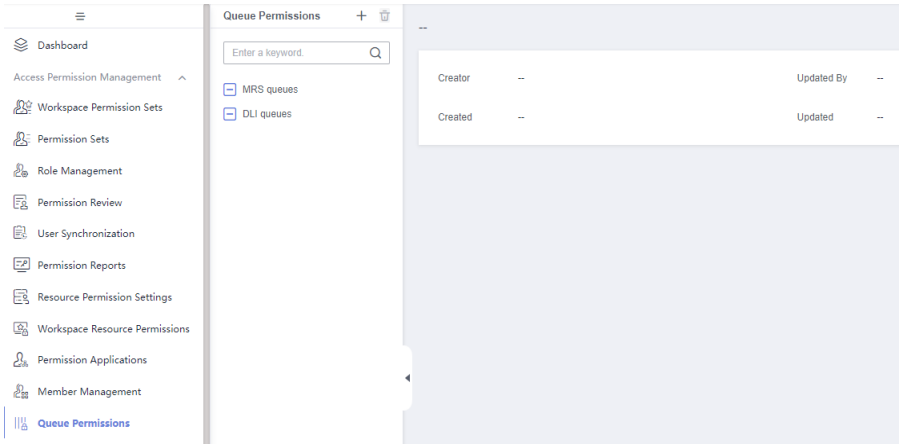
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-104 DataArts Security



Step 2 In the left navigation pane, choose **Queue Permissions**.

Figure 9-105 Queue Permissions page

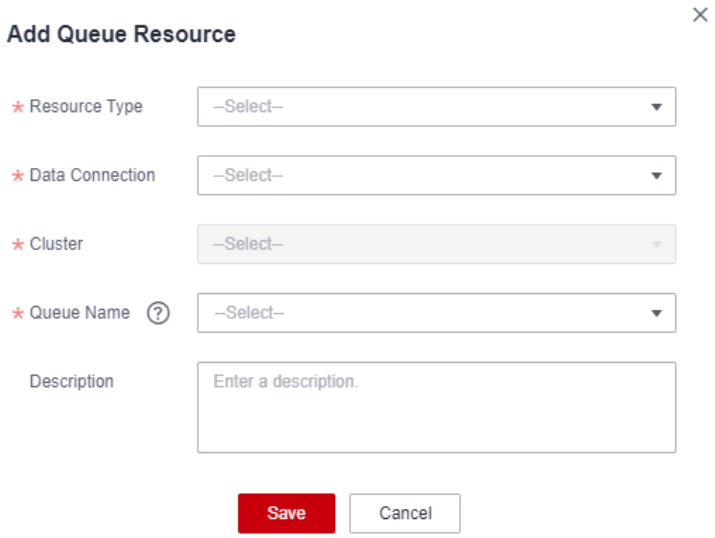


Step 3 Click + above the queue permission directory to allocate a queue to the current workspace. In the displayed **Add Queue Resource** dialog box, set the parameters listed in **Table 9-13** and click **Save**.

Table 9-13 Parameters for adding a queue

Parameter	Description
*Resource Type	Select **MRS queues** or **DLI queues**.
*Data Connection	Select the data connection where the queue is located. For details about how to create a data connection, see **Creating a Data Connection**.
*Cluster Name	This parameter is displayed only when **Resource Type** is set to **MRS queues**. The system automatically matches the cluster name corresponding to the data connection.
*Queue Name	Select the queue to be authorized. • If you set **Resource Type** to **MRS queues**, the available queues are from an MRS cluster. To view the available queues, go to the MRS console, click a cluster name to go to the cluster details page, and click the **Tenants** and then **Queue Configuration** tab. • If you set **Resource Type** to **DLI queues**, the available queues are the queues purchased in DLI. To view the available queues, go to the DLI console and choose **Resources** > **Queue Management**. In addition, DLI queues are classified into SQL queues and general-purpose queues. SQL queues are used to run SQL jobs, and general-purpose queues are used to run Flink and Spark JAR jobs.
Description	Information to make the queue easier to be identified

Figure 9-106 Adding queues



Step 4 Click a queue in the queue permission directory to go to the queue details page.

You can configure attributes for MRS Yarn queues, which are mainly used for task management in DataArts Factory. Real-time queues are used to run real-time jobs, and offline queues are used to run batch jobs. By default, job types of queues are not distinguished.

Figure 9-107 MRS Yarn queue details

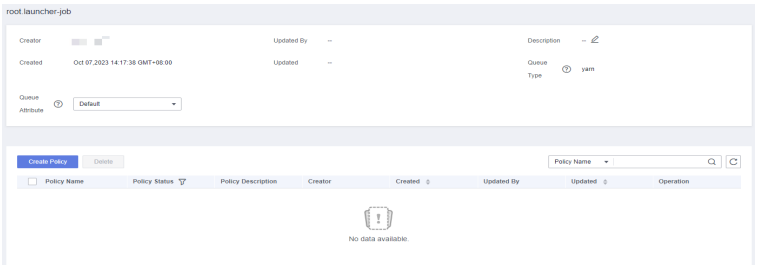
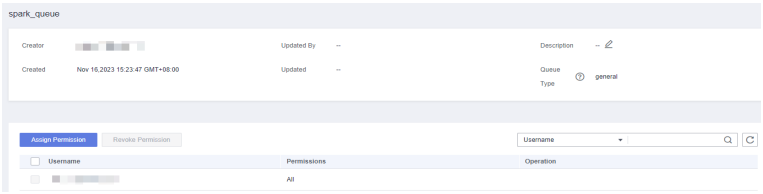


Figure 9-108 DLI queue details



Step 5 Grant permissions to the allocated queues.

- **MRS Yarn queue**

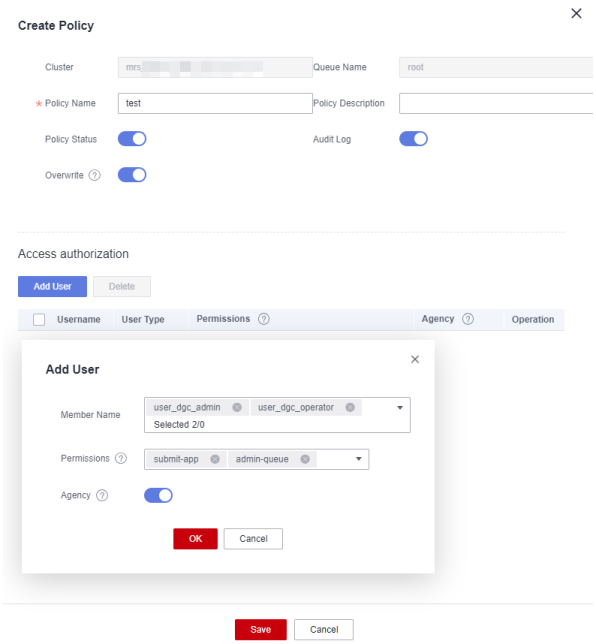
  On the MRS Yarn queue details page, click **Create Policy**. In the displayed dialog box, set the parameters in **Table 9-14** and click **Save**.

  Table 9-14 MRS Yarn queue policy parameters

Parameter	Description
Cluster Name	The system automatically sets this parameter to the name of the cluster where the queue is located.
Queue Name	The system automatically sets this parameter to the current queue name.
*Policy Name	Name of the permission policy for the MRS Yarn queue. To facilitate policy management, you are advised to include the authorization object in the name.
Policy Description	Information to make the policy easier to be identified
Policy Status	If this function is enabled, the current policy takes effect.
Audit Log	If this function is enabled, operation logs of the current queue can be recorded. You can view the audit logs in the data source.

Parameter	Description
Overwrite	Due to the restrictions of the Ranger component, if a queue permission policy already exists for the user or user group in the Ranger component, the current policy may be considered duplicate and cannot be added. If this function is enabled, the system attempts to overwrite the existing queue permission policy for the user or user group in Ranger. If the overwriting fails, you need to delete the queue permission policy of the user or user group from the Ranger component and add the policy again.
***Access Authorization** (Click **Add User** to open the configuration window.)	
Username	Select the users or user groups to be authorized. The users and user groups that have been added to the workspace are available for selection.
Permission	– **submit-app**: the permission required for submitting queues – admin-queue: the permission required for managing queues
Agency	If you want the users or user groups to be authorized to manage this policy, you can enable this option so that the users or user groups become the administrators of this policy and can update or delete the policy.

Figure 9-109 MRS Yarn queue details



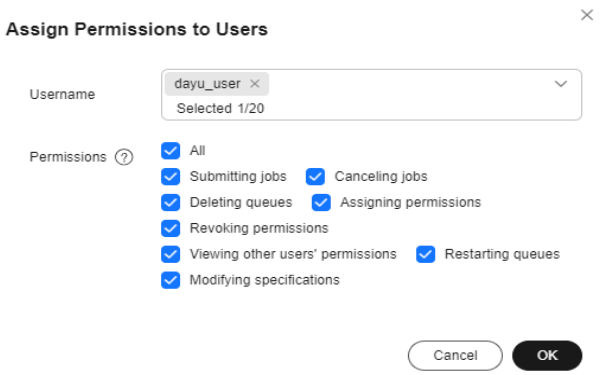
- **DLI queue**

On the DLI queue details page, click **Authorize**. In the displayed dialog box, set the parameters in **Table 9-14** and click **Save**.

Table 9-15 DLI queue authorization parameters

Parameter	Description
Username	Select the users to be authorized. The users that have been added to the workspace are available for selection. **NOTE** Permissions of DLI queues can be granted only to users, but not to user groups.
Permissions	– **Submitting jobs**: This permission allows you to submit jobs to this queue. – **Terminating jobs**: This permission allows you to terminate jobs submitted to this queue. – **Deleting queues**: This permission allows you to delete the queue. – **Granting permissions**: This permission allows you to grant queue permissions to other users. – **Revoking permissions**: This permission allows you to revoke the queue permissions from other users except the queue owner. – **Viewing other users' permissions**: This permission allows you to view the queue permissions of other users. – **Restarting queues**: This permission allows you to restart the queue. – **Modifying queue specifications**: This permission allows you to modify queue specifications.

Figure 9-110 DLI queue details



----End

Related Operations

- Deleting queues: In the queue permission directory, select queues and click to delete them.

  NOTE

  - When a queue is deleted, it is not directly deleted from MRS or DLI. Instead, the queue will no longer be allocated to the workspace.
  - After a queue is deleted, the permissions configured for the queue are still valid. For how to delete queue permissions, see **Deleting policies** or **Revoking permissions**.
  - Yarn queues that are being used in DataArts Factory cannot be deleted in DataArts Security.

- Editing policies: On the MRS Yarn queue details page, locate a policy and click **Edit** in the **Operation** column to edit the policy.
- Deleting policies: On the MRS Yarn queue details page, locate a policy and click **Delete** in the **Operation** column to delete the policy. To delete multiple policies at a time, select the policies and click **Delete** above the policy list.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.

- Modifying permissions: On the DLI queue details page, locate a permission and click **Modify** in the **Operation** column.
- Revoking permissions: On the DLI queue details page, locate a permission and click **Revoke** in the **Operation** column.

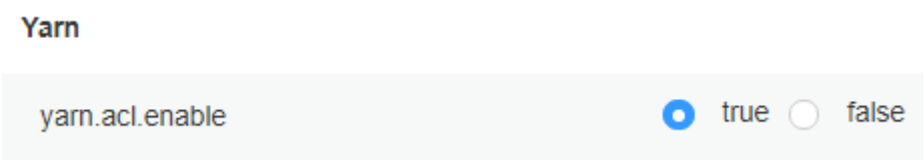
Reference: Configuring Strict Permission Control for Yarn

- The procedure is as follows:

  a. Log in to FusionInsight Manager and choose **Cluster** > **Services** > **Yarn**.

  b. On the displayed page, click the **Configuration** tab then the **All Configurations** sub-tab. On this sub-tab page, search for the **yarn.acl.enable** parameter and change its value to **true**. If the value is **true**, no further action is required.

  Figure 9-111 Configuring yarn.acl.enable

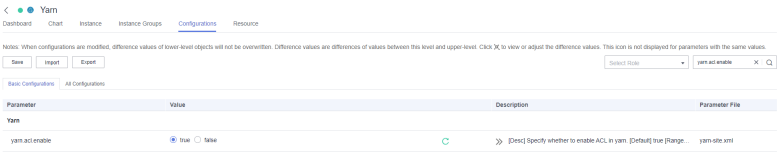
  

Before configuring permissions for Yarn queues, you need to enable permission control for Yarn queues.

Step 1 Log in to MRS FusionInsight Manager.

Step 2 Choose **Cluster** > **Services** > **Yarn** and click **Configurations** and then **Basic Configurations**. Search for the **yarn.acl.enable** parameter and change its value to **true**. If the value is **true**, no further action is required.

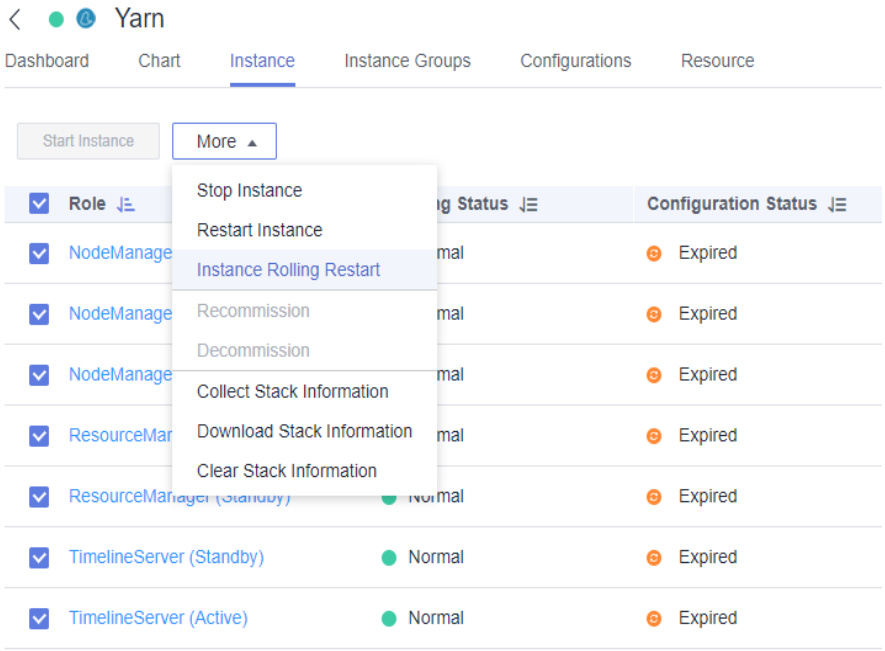
Figure 9-112 Configuring the yarn.acl.enable parameter



Step 3 After the parameter is set, click **Save** in the upper left corner and then **OK** in the dialog box to save the configuration.

Step 4 After saving the configuration, switch to the **Instances** tab page, select the instance that has expired, click **More**, and select **Instance Rolling Restart** to make the configuration take effect.

Figure 9-113 Performing a rolling instance restart



----End

9.3.6.2 Configuring Workspace Resource Permission Policies

This section describes how to use workspace resource permission policies to implement refined permission control on all the data connections and IAM agencies (only those whose agency object is DGC) in the Management Center based on users, user groups, or roles.

- If no workspace resource permission policy is configured for a resource, all users can view and use the resource by default.
- If the permissions of a resource (such as a connection or an agency) are granted to any user, user group, or role, other common users cannot view and use the resource, except the DAYU Administrator, Tenant Administrator, data security administrator, and preset workspace administrator.

Prerequisites

Only the DAYU Administrator, Tenant Administrator, data security administrator, and preset workspace administrator have the permission to create, edit, or delete workspace resource permission policies.

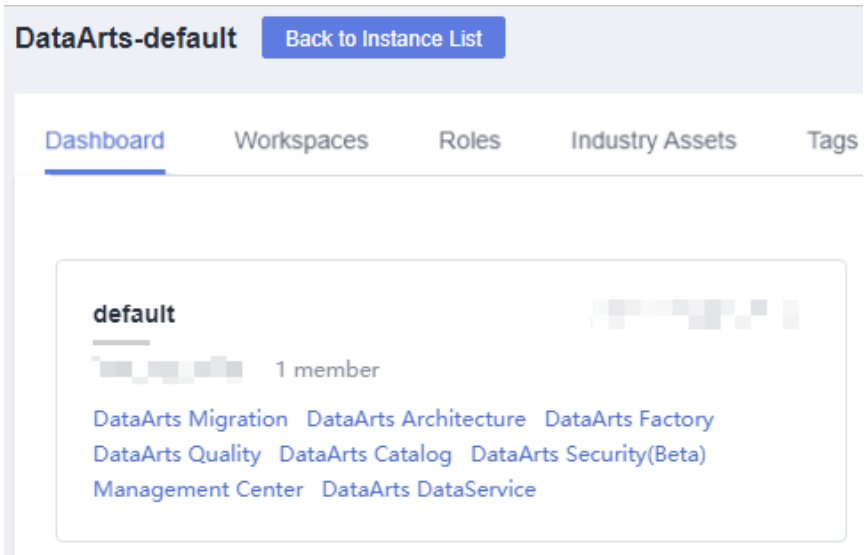
Constraints

- Resources of workspaces in simple mode can be managed, but those of workspaces in enterprise mode cannot.
- If no permission is assigned to a resource, permission control is disabled for the resource.
- Currently, only the DataArts Factory component supports workspace resource permission policies. Other components are not restricted by workspace resource permission policies. In the following scenarios of DataArts Factory, authentication is performed based on workspace resource permission policies:
  - Selecting a connection, job agency, or public agency during script or job development
  - Submitting a script or job
- Resource permission management is not supported for the data connections created in DataArts Factory in earlier versions.
- When a workspace resource is deleted, the corresponding workspace resource permission policy will not be automatically deleted.

Creating a Workspace Resource Permission Policy

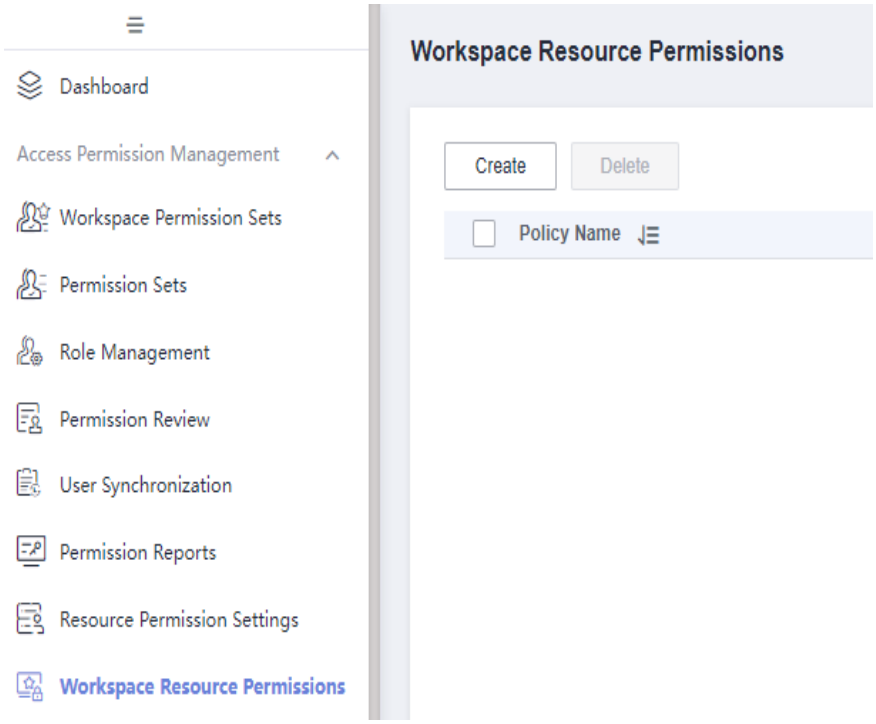
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-114 DataArts Security



Step 2 In the left navigation pane, choose **Workspace Resource Permissions**.

Figure 9-115 Workspace Resource Permissions



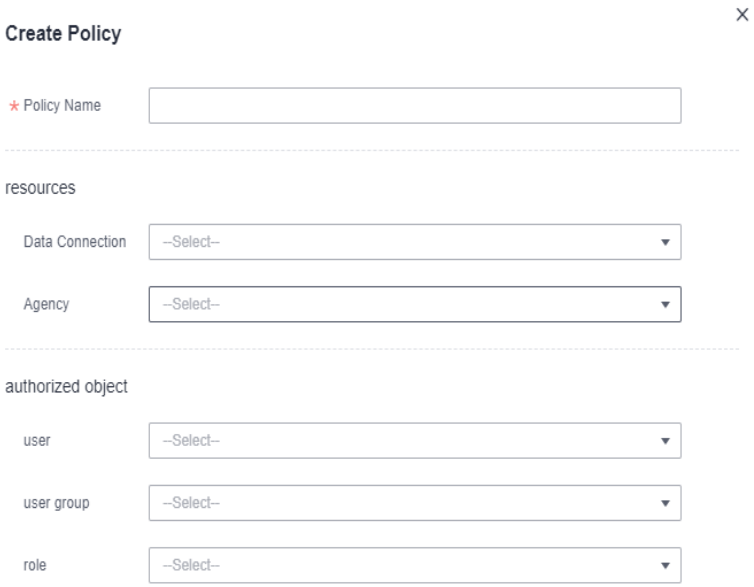
Step 3 On the **Workspace Resource Permissions** page, click **Create**. In the slide-out panel, set the parameters listed in **Table 9-16** and click **Save**.

Table 9-16 Parameters for creating a workspace resource permission policy

Parameter	Description
*Policy Name	Enter the name of the workspace resource permission policy. To facilitate policy management, you are advised to include the resource object and authorization object in the name.
Resource Object	
Data Connection	Select the data connections in the Management Center to be authorized. For details about how to create a data connection, see **Creating a Data Connection**. **NOTE** • Permission control is disabled for the data connections that are not selected. • Unauthorized common users (except the DAYU Administrator, Tenant Administrator, data security administrator, and preset workspace administrator) cannot view or use the selected connections.

Parameter	Description
Agency	Select the IAM agencies to be authorized. Only cloud service agencies whose agency object is DGC are supported. For details about how to create an agency, see **Reference: Creating an Agency**. **NOTE** • Permission control is disabled for the agencies that are not selected. • Unauthorized common users (except the DAYU Administrator, Tenant Administrator, data security administrator, and preset workspace administrator) cannot view or use the selected agencies.
Authorization Object	
user	Select the users to be authorized. The workspace users are available for selection.
user group	Select the user groups to be authorized. The workspace user groups are available for selection.
role	Select the roles to be authorized. The preset and custom roles are available for selection.

Figure 9-116 Creating a workspace resource permission policy



----End

Related Operations

- Editing policies: On the **Workspace Resource Permissions** page, locate a policy and click **Edit** in the **Operation** column to edit the policy.

- Deleting policies: On the **Workspace Resource Permissions** page, locate a policy and click **Delete** in the **Operation** column to delete the policy. To delete multiple policies at a time, select the policies and click **Delete** above the policy list.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.

9.3.7 Controlling Ranger Access Using Permissions

9.3.7.1 Configuring Resource Permissions

This section describes how to create resource permission policies for Ranger to control access to MRS resources and reduce data security risks for your enterprise.

Currently, the following permission policies can be created:

- **Creating an HDFS Permission Policy**
- **Creating a Hive Access Permission Policy**
- **Creating a Hive Masking Permission Policy**
- **Creating a Hive Row-Level Filter Permission Policy**
- **Creating an HBase Permission Policy**
- **Creating a Yarn Permission Policy**
- **Creating a Kafka Permission Policy**
- **Creating a Storm Permission Policy**

Prerequisites

- A Ranger connection has been created in Management Center, and a correct RangerAdmin service IP address and Ranger service port have been set for the connection (see **Configuring an MRS Ranger Connection** for details).

  NOTE

  When you test the Ranger connection in Management Center, the Ranger service IP address and port will not be verified, and no error will be reported even if they are incorrect. You are advised to check them manually.

- Ranger authentication has been enabled for the corresponding MRS cluster. In security mode, Ranger authentication is enabled by default. In common mode, Ranger authentication is disabled by default. For details, see **Enabling Ranger Authentication**.

Constraints

- Resource permission policies depend on the Ranger authentication of MRS clusters. Currently, only permissions on MRS resources can be controlled.
- A permission policy takes effect about 1 minute after being configured.

MRS Components that Support Access Control and the Permission List

Ranger can be used to integrate components in MRS clusters of version 3.0.0 or a later version to enable fine-grained access permission control for components.

Table 9-17 lists the supported components and describes related permissions. For details, see **Configuring Component Permission Policies**.

Table 9-17 Supported components and permissions

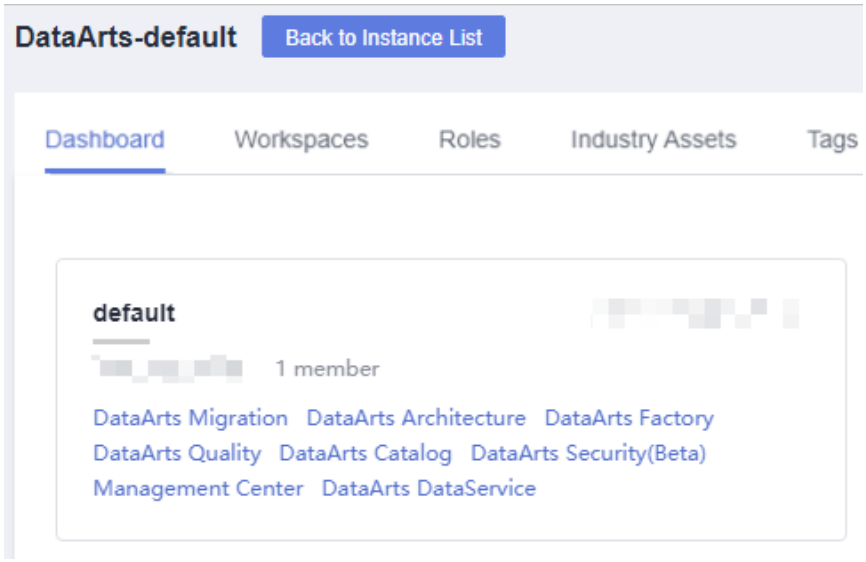
Component	Permission
HDFS	HDFS file permissions: • Read: the permission required for read • Write: the permission required for write • Execute: the permission required for executing a job
Hive	Hive database, data table, and column permissions: • Select: the permission required for query • Update: the permission required for update • Create: the permission required for creation • Drop: the permission required for dropping • Alter: the permission required for alteration • All: the permissions required for all operations • Temporary UDF Admin: the permission required for managing a temporary UDF
Yarn	Yarn queue permissions: • submit-app: the permission required for submitting a queue • admin-queue: the permission required for managing a queue
HBase	HBase column and column family permissions: • Read: the permission required for read • Write: the permission required for write • Create: the permission required for creation • Admin: the permission required by an administrator

Component	Permission
Kafka	Kafka topic permissions: • Publish: the permission required for production • Consume: the permission required for consumption • Configure: the permission required for expanding the capacity of a topic • Describe: the permission required for query • Create: the permission required for creating a topic • Delete: the permission required for deleting a topic • Describe Configs: the permission required for querying configurations • Alter Configs: the permission required for modifying configurations
Storm	Storm topology permissions: • Submit Topology: the permission required for submitting a topology • File Upload: the permission required for uploading a file • File DownLoad: the permission required for downloading a file • Kill Topology: the permission required for deleting a topology • Rebalance: the permission required for rebalance • Activate: the permission required for activation • Deactivate: the permission required for deactivation • Get Topology Conf: the permission required for getting the configurations of a topology • Get Topology: the permission required for getting a topology • Get User Topology: the permission required for getting a user topology • Get Topology Info: the permission required for getting the information of a topology • Upload New Credential: the permission required for uploading a new credential

Creating an HDFS Permission Policy

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-117 DataArts Security

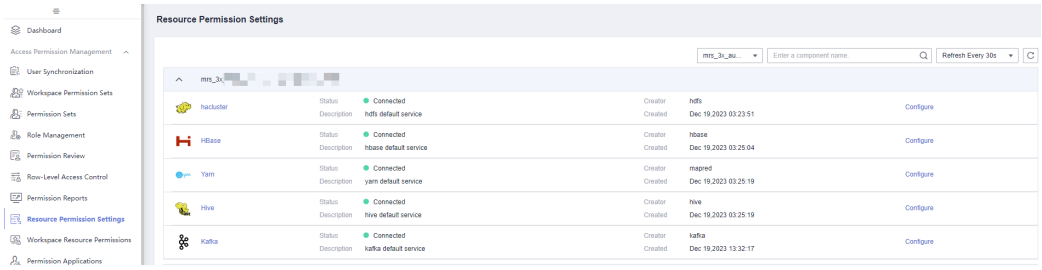


Step 2 Choose **Access Permission Management** > **Resource Permission Settings** from the left navigation bar.

NOTE

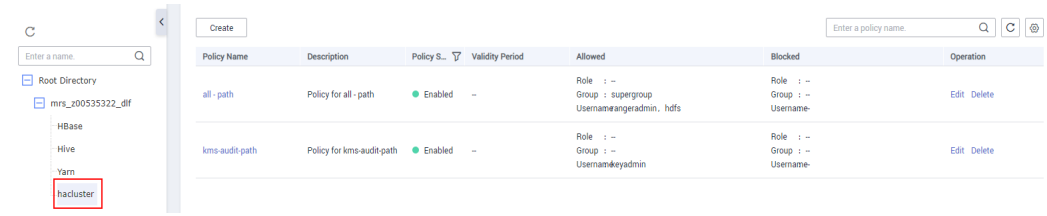
If error message "cluster [mrs_3x_autotest_do_not_del] get service failed. due to [null response from cdm:[404 NOT FOUND]]." is displayed, check whether the RangerAdmin service IP address and Ranger service port of the Ranger connection are correct in Management Center by referring to **Configuring an MRS Ranger Connection**.

Figure 9-118 Resource Permission Settings page



Step 3 Click **Configure** to the right of **hacluster** under the HDFS component, and click **Create** in the upper part of the page displayed.

Figure 9-119 Creating a permission policy



Step 4 Set the parameters and click **OK**.

Figure 9-120 Assigning a permission policy

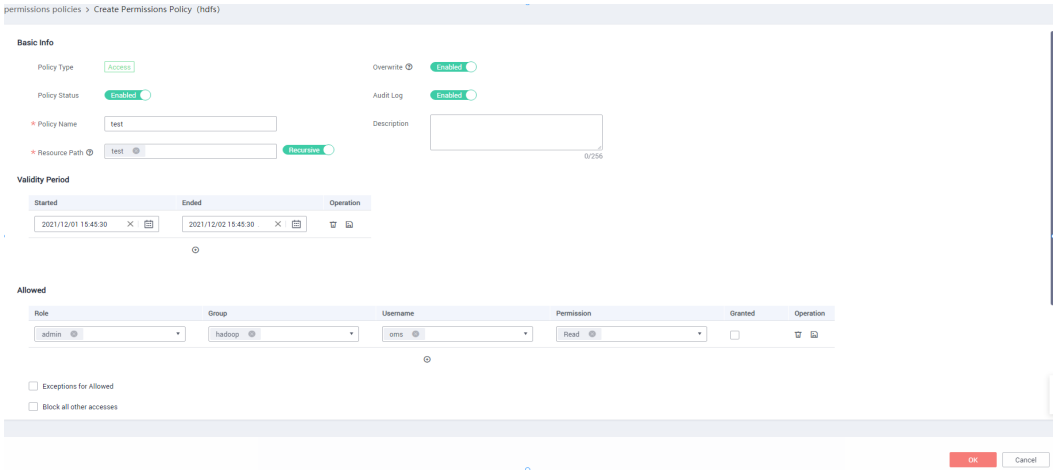


Table 9-18 Parameters for configuring an HDFS permission policy

Parameter	Description
Policy Type	The policy type is automatically generated based on the selected service component. **Policy Type** can be set to **Access**, **Mask**, and **Row-level Filter**. **Mask** and **Row-level Filter** are specific to Hive.
Policy Status	If **Policy Status** is **Enabled**, the permission policy takes effect immediately. If **Policy Status** is **Disabled**, the permission policy does not take effect after being created. **Policy Status** is set to **Enabled** by default.
Overwrite	If **Overwrite** is set to **Enabled**, the new policy takes effect and the old policy does not take effect. **Overwrite** is set to **Enabled** by default. When you need to create a temporary access policy, enable **Overwrite** and set **Validity Period** as required. In this way, even if the temporary access policy expires, the original permission policy still takes effect.
Audit Log	If **Audit Log** is set to **Enabled**, logs are recorded. The log content includes the client access time, client IP address, client user, and resource operation result.
Policy Name	Policy name is mandatory. A policy name can include only letters, numbers, underscores (_), and hyphens (-). Up to 50 characters are allowed.
Description	A description of the policy. Up to 256 characters are allowed.
Topology	HDFS path for access permission control.
Recursion	If the function is enabled, the resource path is in recursive mode. If the function is disabled, the resource path is in non-recursive mode. **Policy Status** is set to **Enabled** by default.

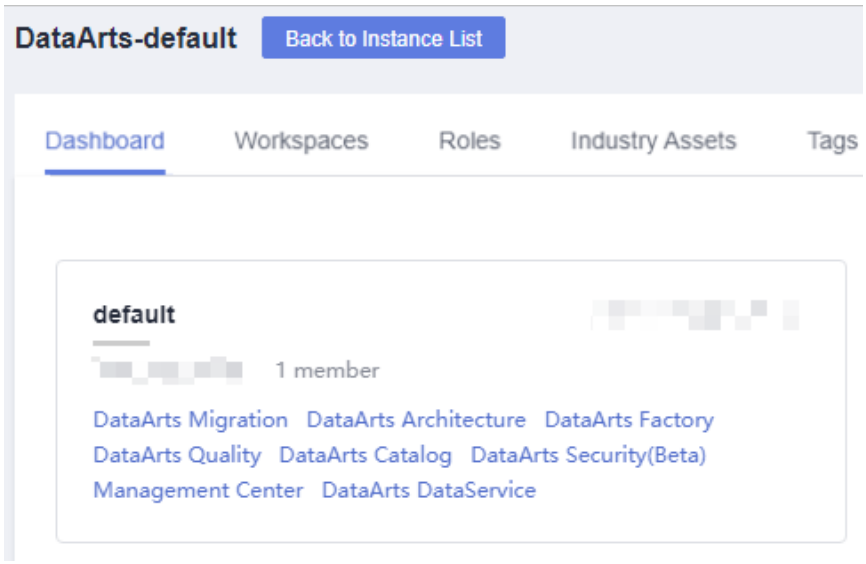
Parameter	Description
Validity Period	You can set the effective time and expiration time of a policy. You can configure multiple time ranges.
Allowed	Users and user groups that are allowed to access the resources. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Granted**: If **Granted** is selected, management permissions are assigned to appropriate users and groups. Delegated administrators can update and delete policies and create sub-policies based on the original policies.
Exceptions	If you select **Exceptions for Allowed**, users who are not allowed to access the system are added to the user group that is allowed to access the system. If you select **Exceptions for blocked**, users who are allowed to access the system are added to the user group that is blocked from the system.
Block all other accesses	If you select **Block all other accesses**, only specified users or user groups are allowed to access the system.
Blocked	**Blocked** is displayed when **Block all other accesses** is not selected. Users and user groups that are not allowed to access the system can be specified in the **Blocked** area. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are not allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Granted**: If **Granted** is selected, management permissions are assigned to appropriate users and groups. Delegated administrators can update and delete policies and create sub-policies based on the original policies.

----End

Creating a Hive Access Permission Policy

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-121 DataArts Security

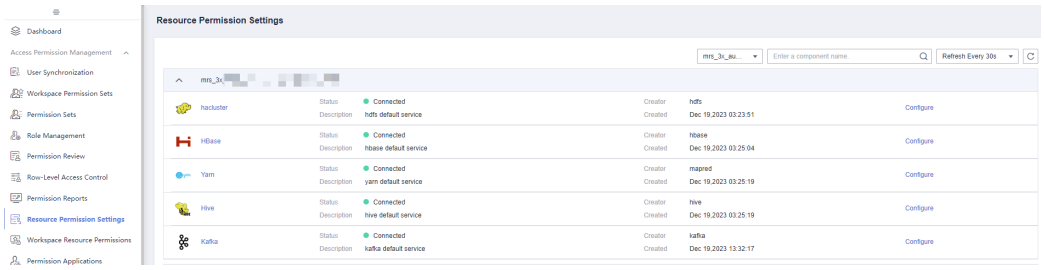


Step 2 Choose **Access Permission Management** > **Resource Permission Settings** from the left navigation bar.

NOTE

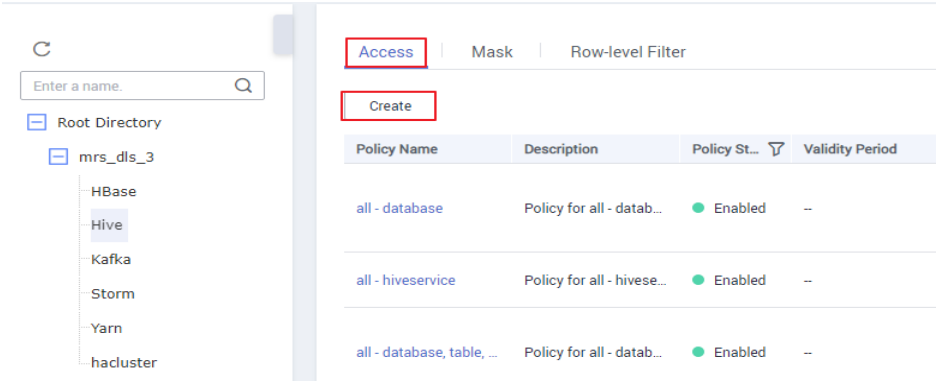
If error message "cluster [mrs_3x_autotest_do_not_del] get service failed. due to [null response from cdm:[404 NOT FOUND]]." is displayed, check whether the RangerAdmin service IP address and Ranger service port of the Ranger connection are correct in Management Center by referring to **Configuring an MRS Ranger Connection**.

Figure 9-122 Resource Permission Settings page



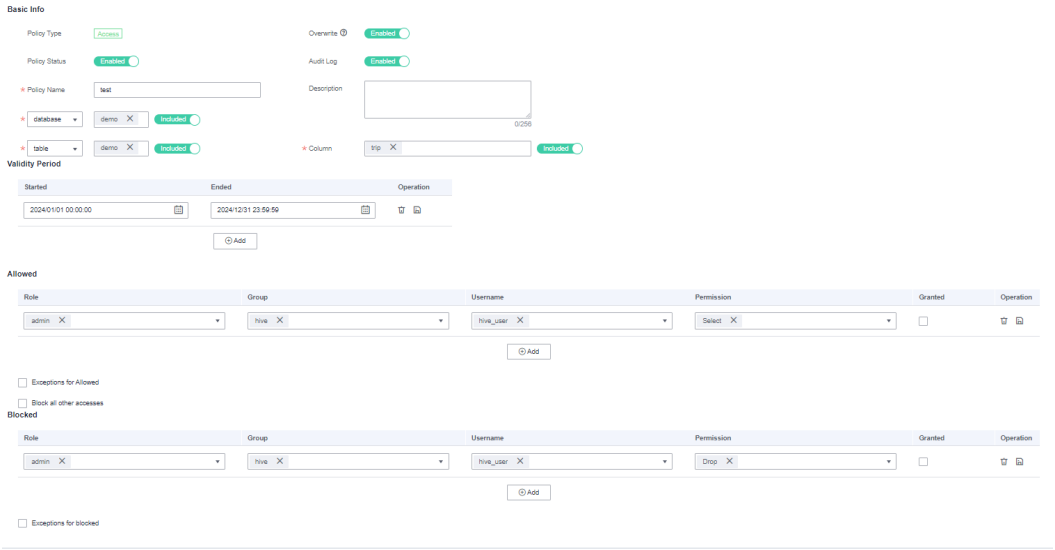
Step 3 Click **Configure** to the right of the Hive component, and click **Create** in the upper part of the page displayed.

Figure 9-123 Creating a permission policy



Step 4 Set the parameters and click **OK**.

Figure 9-124 Configuring a Hive policy



The following table lists the parameters of a Hive permission policy.

Table 9-19 Parameters of a Hive permission policy

Parameter	Description
Policy Type	The policy type is automatically generated based on the selected service component. **Policy Type** can be set to **Access**, **Mask**, and **Row-level Filter**. **Mask** and **Row-level Filter** are specific to Hive.
Policy Status	If **Policy Status** is **Enabled**, the permission policy takes effect immediately. If **Policy Status** is **Disabled**, the permission policy does not take effect after being created. **Policy Status** is set to **Enabled** by default.

Parameter	Description
Overwrite	If **Overwrite** is set to **Enabled**, the new policy takes effect and the old policy does not take effect. **Overwrite** is set to **Enabled** by default. To create a temporary access policy, enable **Overwrite** and set **Validity Period** as required. In this way, even if the temporary access policy expires, the original permission policy still takes effect.
Audit Log	If **Audit Log** is set to **Enabled**, logs are recorded. The log content includes the client access time, client IP address, client user, and resource operation result.
Policy Name	Policy name is mandatory. A policy name can include only letters, numbers, underscores (_), and hyphens (-). Up to 50 characters are allowed.
Description	A description of the policy. Up to 256 characters are allowed.
database	The **database** parameter is mandatory. You can set the database whose permissions need to be controlled. Fuzzy search is supported.
table	The **table** parameter is mandatory. You can set the table whose permissions need to be controlled. Fuzzy search is supported.
Column	The **Column** parameter is mandatory. You can set the column whose permissions need to be controlled. Fuzzy search is supported.
Validity Period	You can set the effective time and expiration time of a policy. You can configure multiple time ranges.
Allowed	Users and user groups that are allowed to access the resources. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Granted**: If **Granted** is selected, management permissions are assigned to appropriate users and groups. Delegated administrators can update and delete policies and create sub-policies based on the original policies.

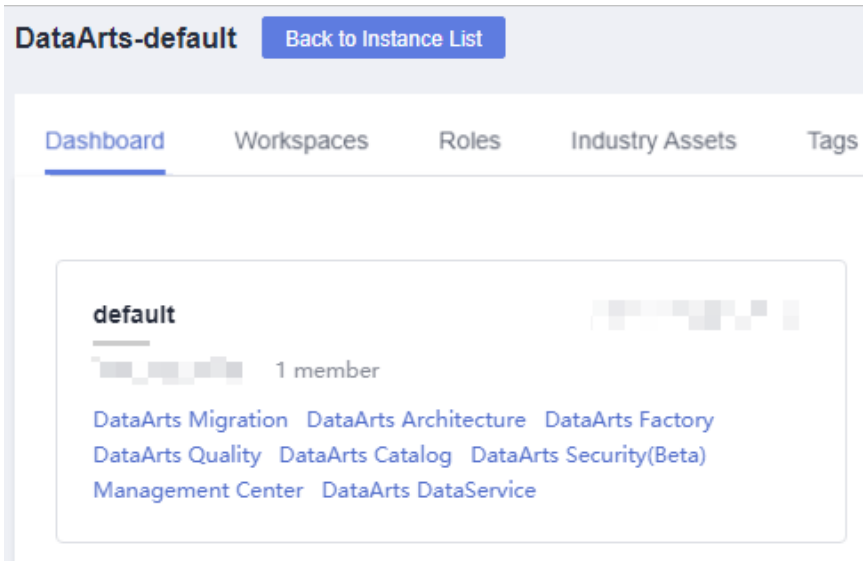
Parameter	Description
Exceptions	If you select **Exceptions for Allowed**, users who are not allowed to access the system are added to the user group that is allowed to access the system. If you select **Exceptions for blocked**, users who are allowed to access the system are added to the user group that is blocked from the system.
Block all other accesses	If you select **Block all other accesses**, only specified users or user groups are allowed to access the system.
Blocked	**Blocked** is displayed when **Block all other accesses** is not selected. Users and user groups that are not allowed to access the system can be specified in the **Blocked** area. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are not allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Granted**: If **Granted** is selected, management permissions are assigned to appropriate users and groups. Delegated administrators can update and delete policies and create sub-policies based on the original policies.

----End

Creating a Hive Masking Permission Policy

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-125 DataArts Security

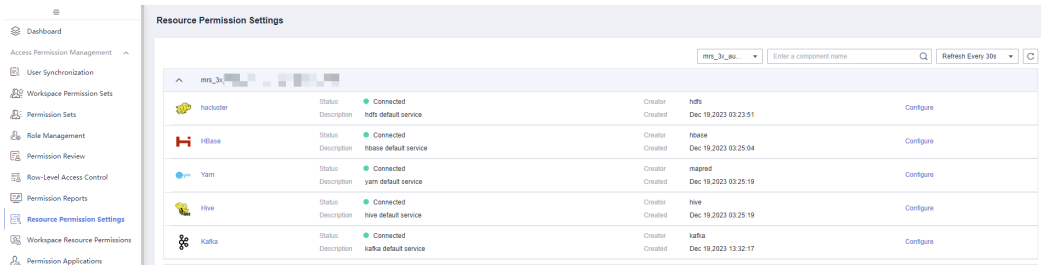


Step 2 Choose **Access Permission Management** > **Resource Permission Settings** from the left navigation bar.

NOTE

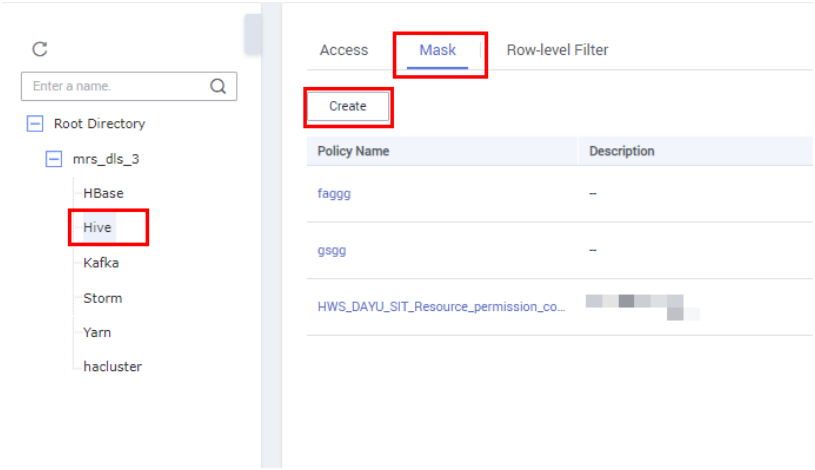
If error message "cluster [mrs_3x_autotest_do_not_del] get service failed. due to [null response from cdm:[404 NOT FOUND]]." is displayed, check whether the RangerAdmin service IP address and Ranger service port of the Ranger connection are correct in Management Center by referring to **Configuring an MRS Ranger Connection**.

Figure 9-126 Resource Permission Settings page



Step 3 Click **Configure** to the right of the Hive component, and click **Create** in the upper part of the **Mask** tab page.

Figure 9-127 Creating a permission policy



Step 4 Set the parameters and click **OK**.

Figure 9-128 Configuring a Hive policy

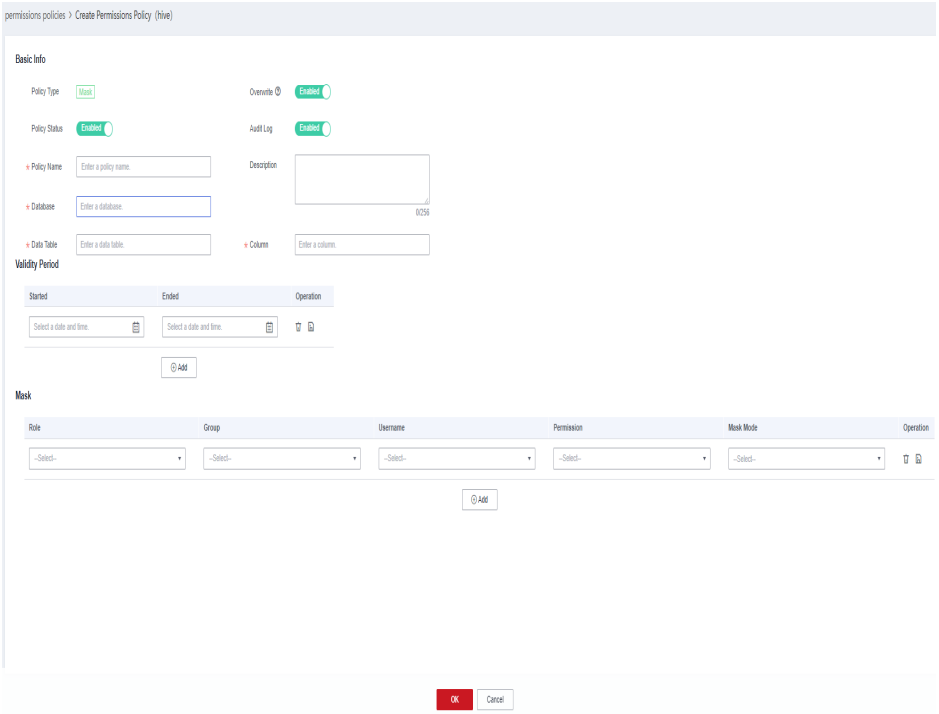


Table 9-20 Parameters of a Hive permission policy

Parameter	Description
Policy Type	The policy type is automatically generated based on the selected service component. **Policy Type** can be set to **Access**, **Mask**, and **Row-level Filter**. **Mask** and **Row-level Filter** are specific to Hive.

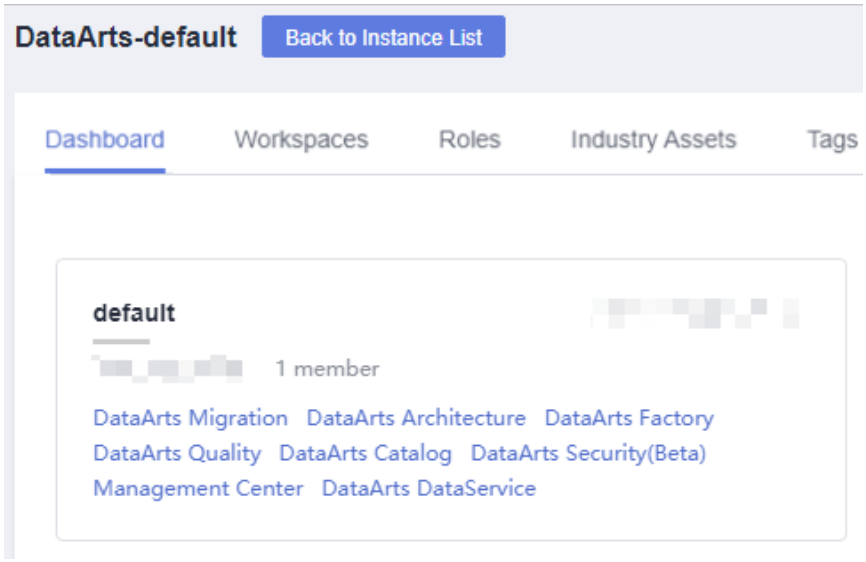
Parameter	Description
Policy Status	If **Policy Status** is **Enabled**, the permission policy takes effect immediately. If **Policy Status** is **Disabled**, the permission policy does not take effect after being created. **Policy Status** is set to **Enabled** by default.
Overwrite	If **Overwrite** is set to **Enabled**, the new policy takes effect and the old policy does not take effect. **Overwrite** is set to **Enabled** by default. To create a temporary access policy, enable **Overwrite** and set **Validity Period** as required. In this way, even if the temporary access policy expires, the original permission policy still takes effect.
Audit Log	If **Audit Log** is set to **Enabled**, logs are recorded. The log content includes the client access time, client IP address, client user, and resource operation result.
Policy Name	Policy name is mandatory. A policy name can include only letters, numbers, underscores (_), and hyphens (-). Up to 50 characters are allowed.
Description	A description of the policy. Up to 256 characters are allowed.
Database	The **Database** parameter is mandatory. You can set the database whose permissions need to be controlled. Fuzzy search is supported.
Data Table	The **Data Table** parameter is mandatory. You can set the table whose permissions need to be controlled. Fuzzy search is supported.
Column	The **Column** parameter is mandatory. You can set the column whose permissions need to be controlled. Fuzzy search is supported.
Validity Period	You can set the effective time and expiration time of a policy. You can configure multiple time ranges.
Mask	The masking mode for users or user groups to access data. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Mask Mode**: Columns that require permission control in a Hive table are masked based on the value of this parameter.

----End

Creating a Hive Row-Level Filter Permission Policy

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-129 DataArts Security

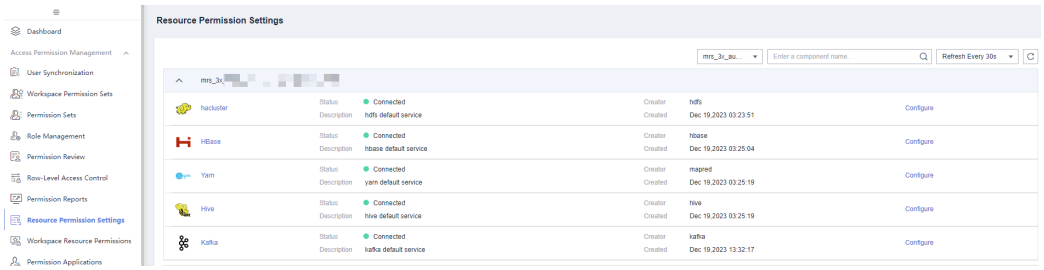


Step 2 Choose **Access Permission Management** > **Resource Permission Settings** from the left navigation bar.

NOTE

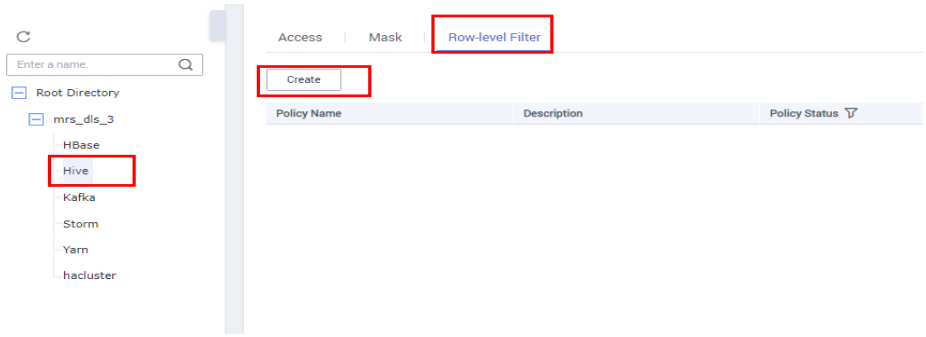
If error message "cluster [mrs_3x_autotest_do_not_del] get service failed. due to [null response from cdm:[404 NOT FOUND]]." is displayed, check whether the RangerAdmin service IP address and Ranger service port of the Ranger connection are correct in Management Center by referring to **Configuring an MRS Ranger Connection**.

Figure 9-130 Resource Permission Settings page



Step 3 Click **Configure** to the right of the Hive component, and click **Create** in the upper part of the **Row-level Filter** tab page.

Figure 9-131 Creating a Hive row-level filter policy



Step 4 Set the parameters and click **OK**.

Figure 9-132 Configuring a Hive policy

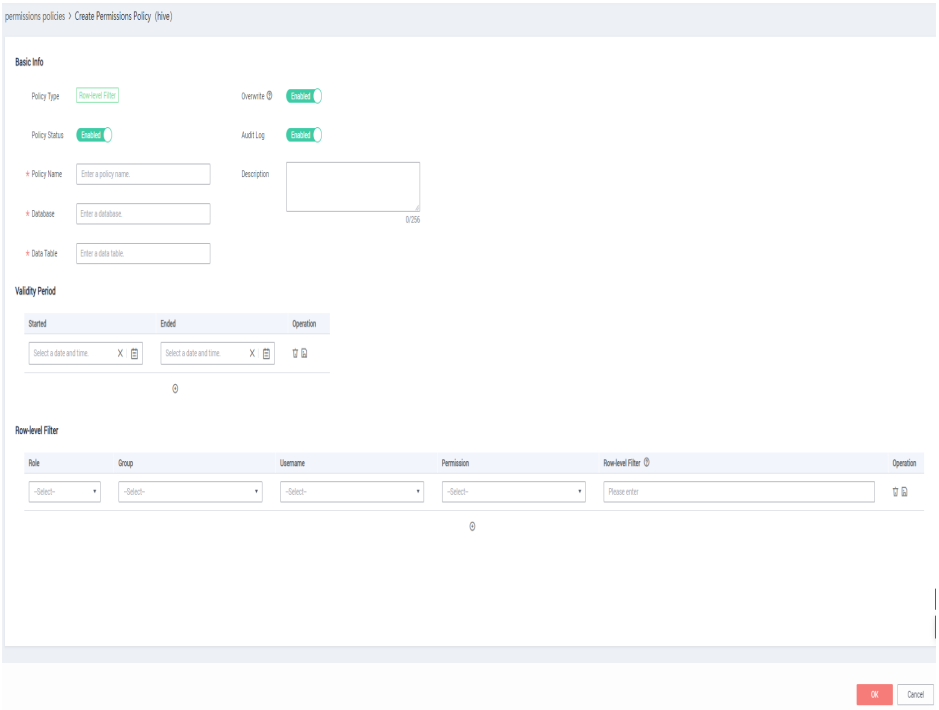


Table 9-21 Parameters of a Hive permission policy

Parameter	Description
Policy Type	The policy type is automatically generated based on the selected service component. **Policy Type** can be set to **Access**, **Mask**, and **Row-level Filter**. **Mask** and **Row-level Filter** are specific to Hive.
Policy Status	If **Policy Status** is **Enabled**, the permission policy takes effect immediately. If **Policy Status** is **Disabled**, the permission policy does not take effect after being created. **Policy Status** is set to **Enabled** by default.

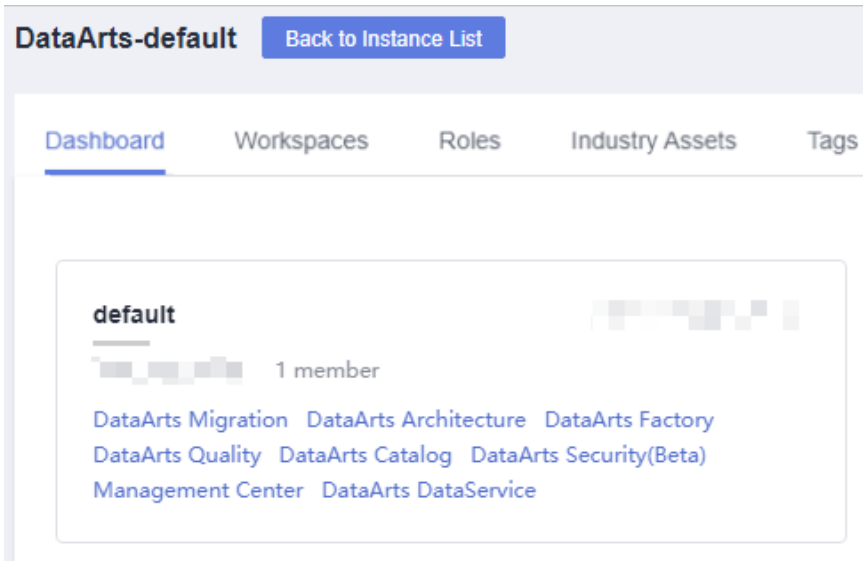
Parameter	Description
Overwrite	If **Overwrite** is set to **Enabled**, the new policy takes effect and the old policy does not take effect. **Overwrite** is set to **Enabled** by default. To create a temporary access policy, enable **Overwrite** and set **Validity Period** as required. In this way, even if the temporary access policy expires, the original permission policy still takes effect.
Audit Log	If **Audit Log** is set to **Enabled**, logs are recorded. The log content includes the client access time, client IP address, client user, and resource operation result.
Policy Name	Policy name is mandatory. A policy name can include only letters, numbers, underscores (_), and hyphens (-). Up to 50 characters are allowed.
Description	A description of the policy. Up to 256 characters are allowed.
Database	The **Database** parameter is mandatory. You can set the database whose permissions need to be controlled. Fuzzy search is supported.
Data Table	The **Data Table** parameter is mandatory. You can set the table whose permissions need to be controlled. Fuzzy search is supported.
Column	The **Column** parameter is mandatory. You can set the column whose permissions need to be controlled. Fuzzy search is supported.
Validity Period	You can set the effective time and expiration time of a policy. You can configure multiple time ranges.
Row-level Filter	Users and user groups that are allowed to access the resources. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Row-level Filter**: Filter by field content. The parameter format is as follows: Field=Value. Example: state=1.

----End

Creating an HBase Permission Policy

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-133 DataArts Security

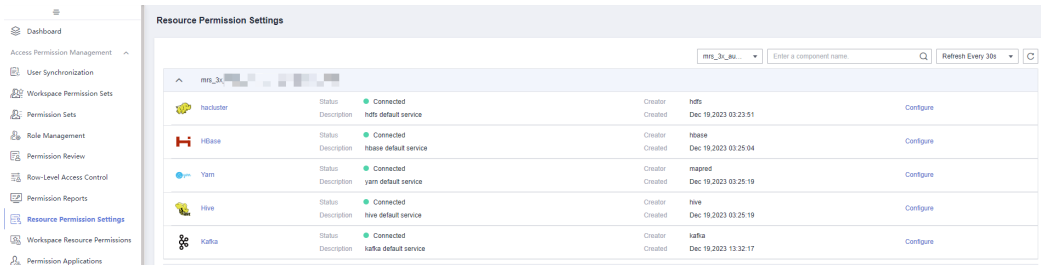


Step 2 Choose **Access Permission Management** > **Resource Permission Settings** from the left navigation bar.

NOTE

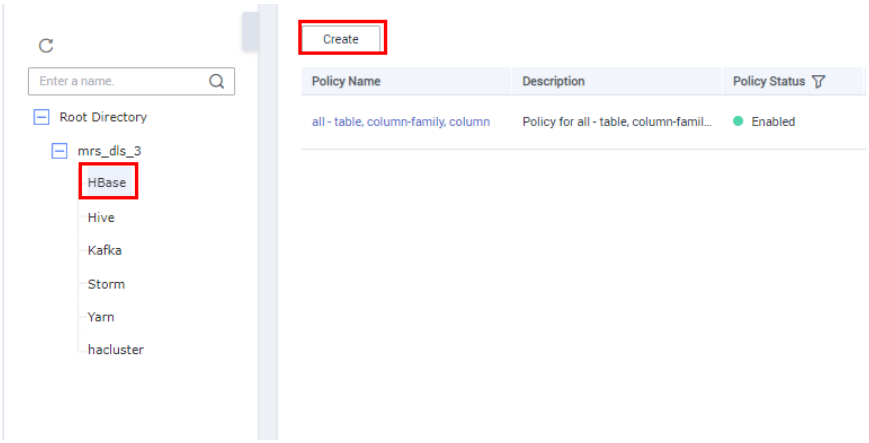
If error message "cluster [mrs_3x_autotest_do_not_del] get service failed. due to [null response from cdm:[404 NOT FOUND]]." is displayed, check whether the RangerAdmin service IP address and Ranger service port of the Ranger connection are correct in Management Center by referring to **Configuring an MRS Ranger Connection**.

Figure 9-134 Resource Permission Settings page



Step 3 Click **Configure** to the right of the HBase component, and click **Create** in the upper part of the page displayed.

Figure 9-135 Creating an HBase permission policy



Step 4 Set the parameters and click **OK**.

Figure 9-136 Configuring an HBase policy

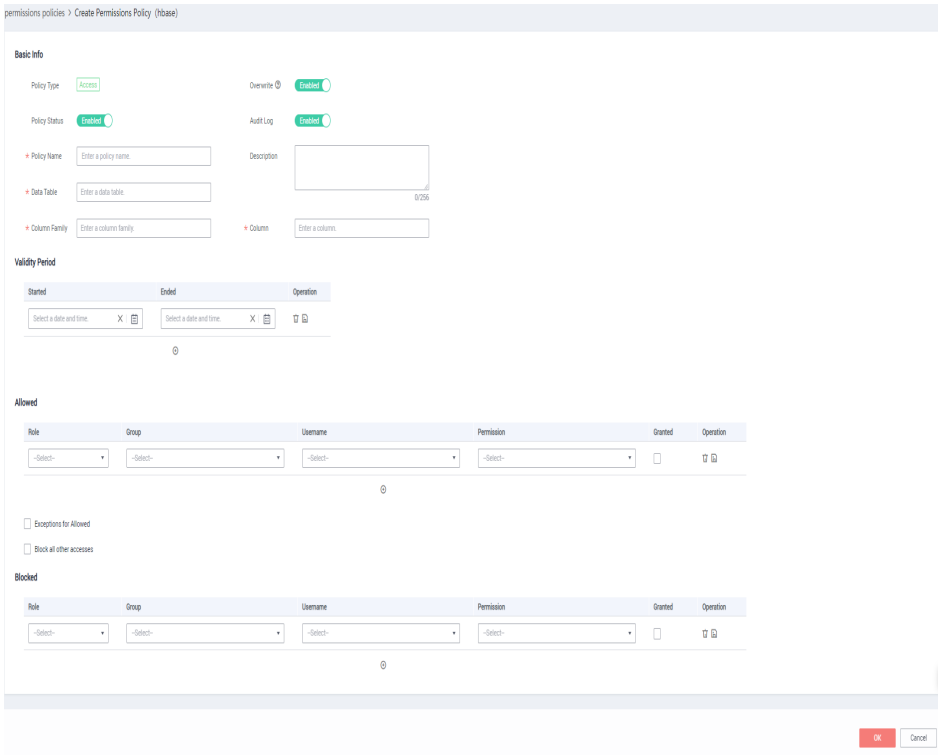


Table 9-22 Parameters of an HBase permission policy

Parameter	Description
Policy Type	The policy type is automatically generated based on the selected service component. **Policy Type** can be set to **Access**, **Mask**, and **Row-level Filter**. **Mask** and **Row-level Filter** are specific to Hive.

Parameter	Description
Policy Status	If **Policy Status** is **Enabled**, the permission policy takes effect immediately. If **Policy Status** is **Disabled**, the permission policy does not take effect after being created. **Policy Status** is set to **Enabled** by default.
Overwrite	If **Overwrite** is set to **Enabled**, the new policy takes effect and the old policy does not take effect. **Overwrite** is set to **Enabled** by default. To create a temporary access policy, enable **Overwrite** and set **Validity Period** as required. In this way, even if the temporary access policy expires, the original permission policy still takes effect.
Audit Log	If **Audit Log** is set to **Enabled**, logs are recorded. The log content includes the client access time, client IP address, client user, and resource operation result.
Policy Name	Policy name is mandatory. A policy name can include only letters, numbers, underscores (_), and hyphens (-). Up to 50 characters are allowed.
Description	A description of the policy. Up to 256 characters are allowed.
Data Table	The **Data Table** parameter is mandatory. You can set the table whose permissions need to be controlled. Fuzzy search is supported.
Column	The **Column** parameter is mandatory. You can set the column whose permissions need to be controlled. Fuzzy search is supported.
Column Family	**Column Family** is mandatory. This parameter indicates a set of column families in an HBase cluster.
Validity Period	You can set the effective time and expiration time of a policy. You can configure multiple time ranges.
Allowed	Users and user groups that are allowed to access the resources. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Granted**: If **Granted** is selected, management permissions are assigned to appropriate users and groups. Delegated administrators can update and delete policies and create sub-policies based on the original policies.

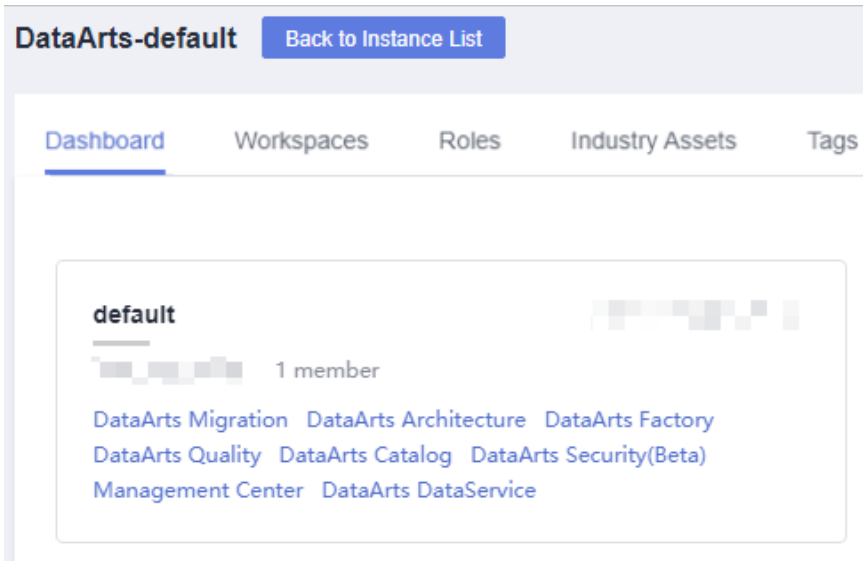
Parameter	Description
Exceptions	If you select **Exceptions for Allowed**, users who are not allowed to access the system are added to the user group that is allowed to access the system. If you select **Exceptions for blocked**, users who are allowed to access the system are added to the user group that is blocked from the system.
Block all other accesses	If you select **Block all other accesses**, only specified users or user groups are allowed to access the system.
Blocked	**Blocked** is displayed when **Block all other accesses** is not selected. Users and user groups that are not allowed to access the system can be specified in the **Blocked** area. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are not allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Granted**: If **Granted** is selected, management permissions are assigned to appropriate users and groups. Delegated administrators can update and delete policies and create sub-policies based on the original policies.

----End

Creating a Yarn Permission Policy

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-137 DataArts Security

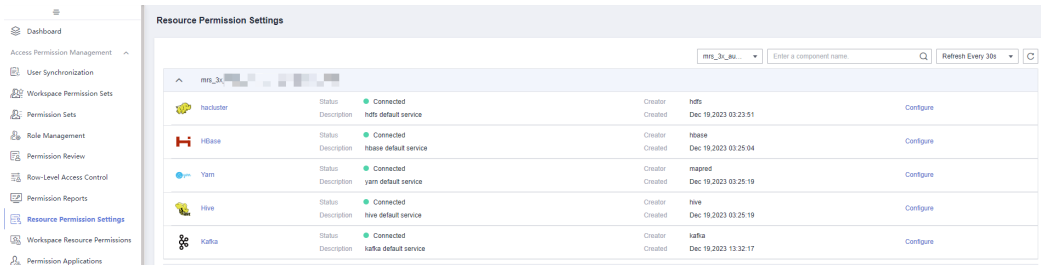


Step 2 Choose **Access Permission Management** > **Resource Permission Settings** from the left navigation bar.

NOTE

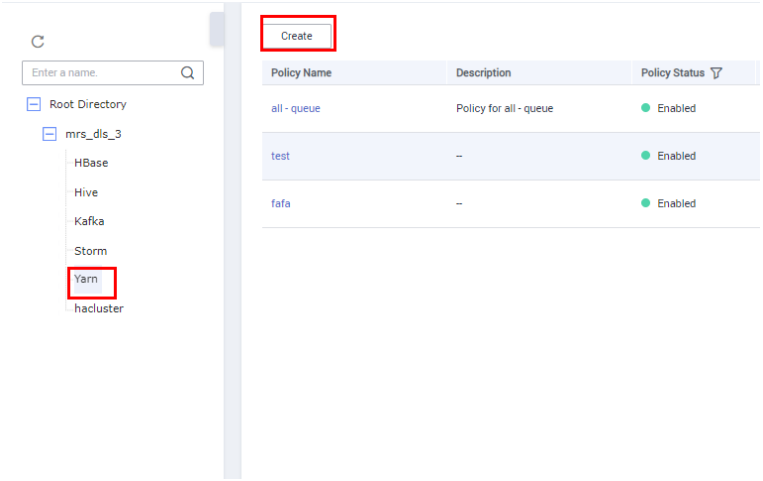
If error message "cluster [mrs_3x_autotest_do_not_del] get service failed. due to [null response from cdm:[404 NOT FOUND]]." is displayed, check whether the RangerAdmin service IP address and Ranger service port of the Ranger connection are correct in Management Center by referring to **Configuring an MRS Ranger Connection**.

Figure 9-138 Resource Permission Settings page



Step 3 Click **Configure** to the right of the Yarn component, and click **Create** in the upper part of the page that is displayed.

Figure 9-139 Creating a Yarn permission policy



Step 4 Set the parameters and click **OK**.

Figure 9-140 Configuring a Yarn policy

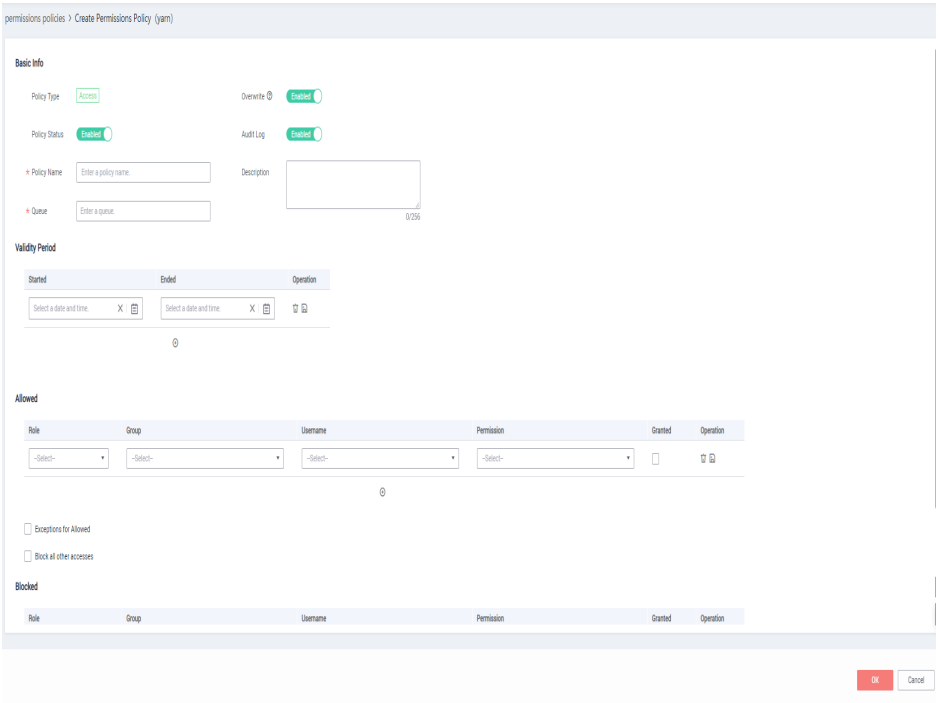


Table 9-23 Parameters of a Yarn permission policy

Parameter	Description
Policy Type	The policy type is automatically generated based on the selected service component. **Policy Type** can be set to **Access**, **Mask**, and **Row-level Filter**. **Mask** and **Row-level Filter** are specific to Hive.

Parameter	Description
Policy Status	If **Policy Status** is **Enabled**, the permission policy takes effect immediately. If **Policy Status** is **Disabled**, the permission policy does not take effect after being created. **Policy Status** is set to **Enabled** by default.
Overwrite	If **Overwrite** is set to **Enabled**, the new policy takes effect and the old policy does not take effect. **Overwrite** is set to **Enabled** by default. To create a temporary access policy, enable **Overwrite** and set **Validity Period** as required. In this way, even if the temporary access policy expires, the original permission policy still takes effect.
Audit Log	If **Audit Log** is set to **Enabled**, logs are recorded. The log content includes the client access time, client IP address, client user, and resource operation result.
Policy Name	Policy name is mandatory. A policy name can include only letters, numbers, underscores (_), and hyphens (-). Up to 50 characters are allowed.
Description	A description of the policy. Up to 256 characters are allowed.
Queue	Resource scheduling queue in the Yarn service.
Validity Period	You can set the effective time and expiration time of a policy. You can configure multiple time ranges.
Allowed	Users and user groups that are allowed to access the resources. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Granted**: If **Granted** is selected, management permissions are assigned to appropriate users and groups. Delegated administrators can update and delete policies and create sub-policies based on the original policies.
Exceptions	If you select **Exceptions for Allowed**, users who are not allowed to access the system are added to the user group that is allowed to access the system. If you select **Exceptions for blocked**, users who are allowed to access the system are added to the user group that is blocked from the system.
Block all other accesses	If you select **Block all other accesses**, only specified users or user groups are allowed to access the system.

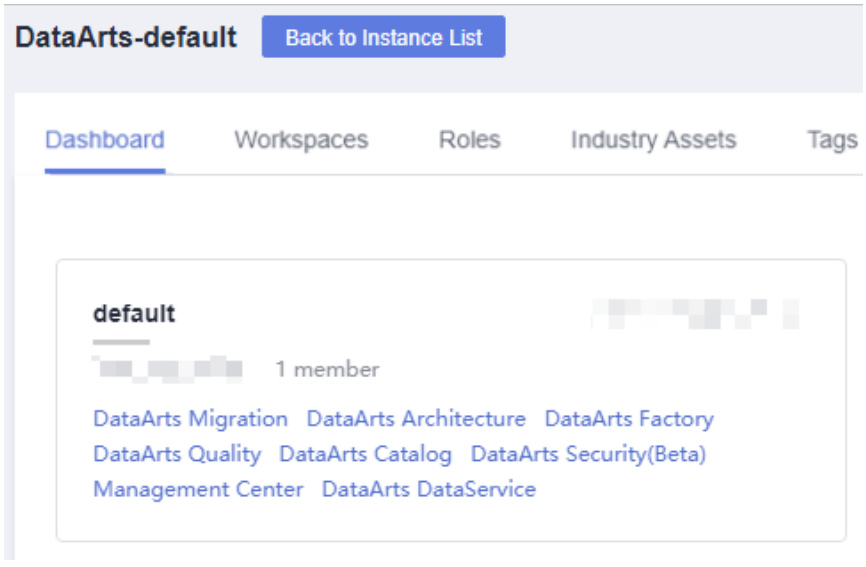
Parameter	Description
Blocked	**Blocked** is displayed when **Block all other accesses** is not selected. Users and user groups that are not allowed to access the system can be specified in the **Blocked** area. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are not allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Granted**: If **Granted** is selected, management permissions are assigned to appropriate users and groups. Delegated administrators can update and delete policies and create sub-policies based on the original policies.

----End

Creating a Kafka Permission Policy

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-141 DataArts Security

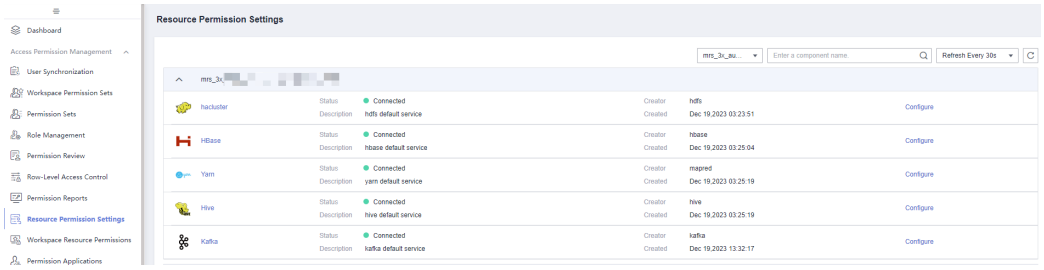


Step 2 Choose **Access Permission Management** > **Resource Permission Settings** from the left navigation bar.

NOTE

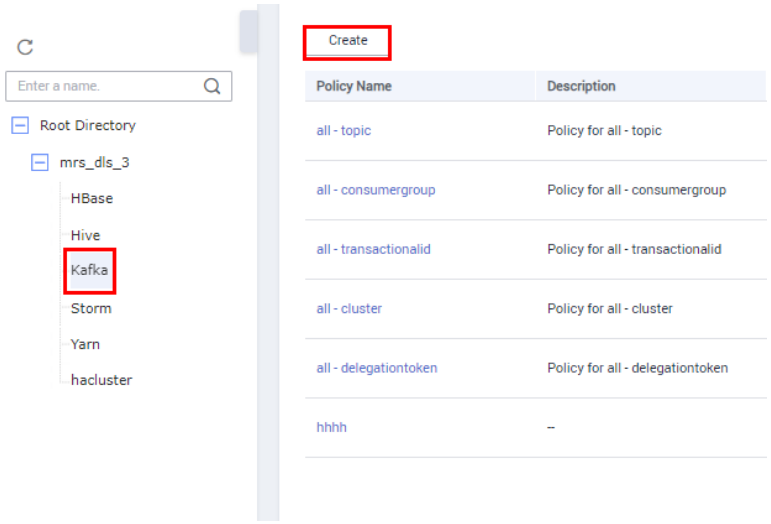
If error message "cluster [mrs_3x_autotest_do_not_del] get service failed. due to [null response from cdm:[404 NOT FOUND]]." is displayed, check whether the RangerAdmin service IP address and Ranger service port of the Ranger connection are correct in Management Center by referring to **Configuring an MRS Ranger Connection**.

Figure 9-142 Resource Permission Settings page



Step 3 Click **Configure** to the right of the Kafka component, and click **Create** in the upper part of the page that is displayed.

Figure 9-143 Creating a Kafka permission policy



Step 4 Set the parameters and click **OK**.

Figure 9-144 Configuring a Kafka policy

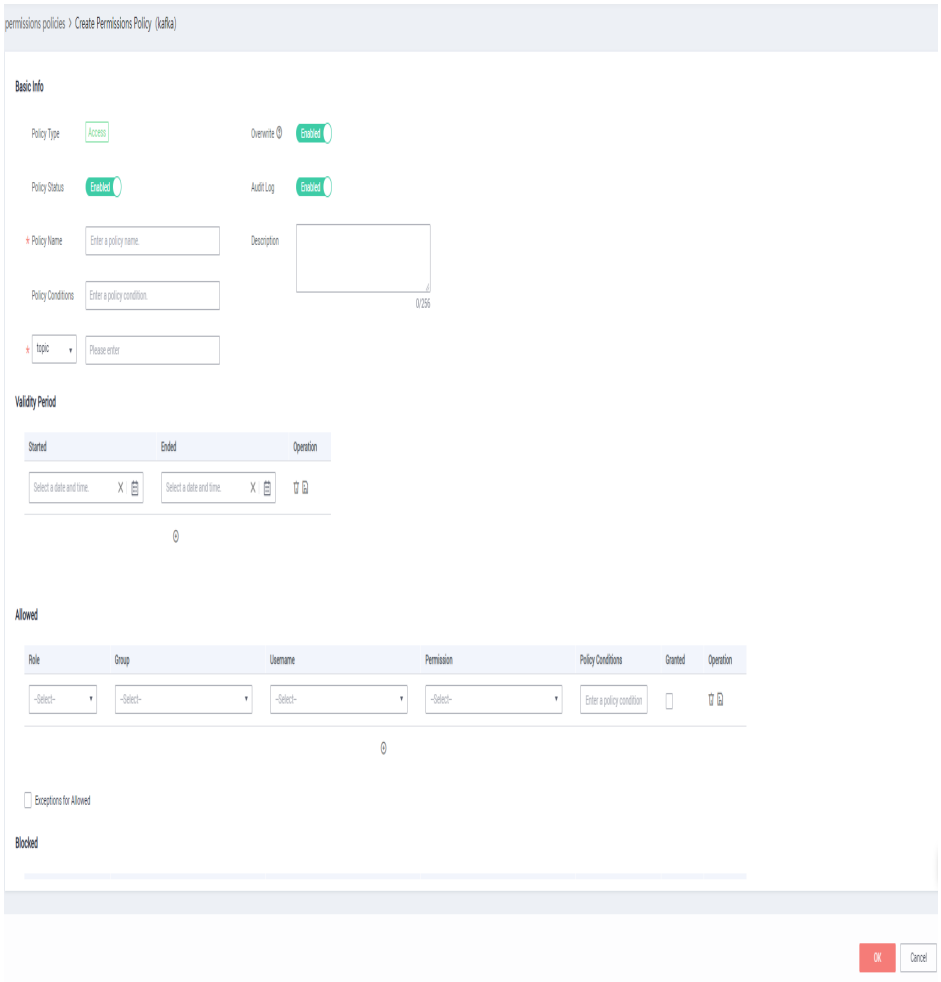


Table 9-24 Parameters of a Kafka permission policy

Parameter	Description
Policy Type	The policy type is automatically generated based on the selected service component. **Policy Type** can be set to **Access**, **Mask**, and **Row-level Filter**. **Mask** and **Row-level Filter** are specific to Hive.
Policy Status	If **Policy Status** is **Enabled**, the permission policy takes effect immediately. If **Policy Status** is **Disabled**, the permission policy does not take effect after being created. **Policy Status** is set to **Enabled** by default.
Overwrite	If **Overwrite** is set to **Enabled**, the new policy takes effect and the old policy does not take effect. **Overwrite** is set to **Enabled** by default. To create a temporary access policy, enable **Overwrite** and set **Validity Period** as required. In this way, even if the temporary access policy expires, the original permission policy still takes effect.

Parameter	Description
Audit Log	If **Audit Log** is set to **Enabled**, logs are recorded. The log content includes the client access time, client IP address, client user, and resource operation result.
Policy Name	Policy name is mandatory. A policy name can include only letters, numbers, underscores (_), and hyphens (-). Up to 50 characters are allowed.
Description	A description of the policy. Up to 256 characters are allowed.
Policy Conditions	Range of IP addresses that can access the Kafka topic.
Topic	The message topic of a Kafka cluster.
Validity Period	You can set the effective time and expiration time of a policy. You can configure multiple time ranges.
Allowed	Users and user groups that are allowed to access the resources. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Policy Conditions**: the range of IP addresses that can access the Kafka topic. • **Granted**: If **Granted** is selected, management permissions are assigned to appropriate users and groups. Delegated administrators can update and delete policies and create sub-policies based on the original policies.
Exceptions	If you select **Exceptions for Allowed**, users who are not allowed to access the system are added to the user group that is allowed to access the system. If you select **Exceptions for blocked**, users who are allowed to access the system are added to the user group that is blocked from the system.

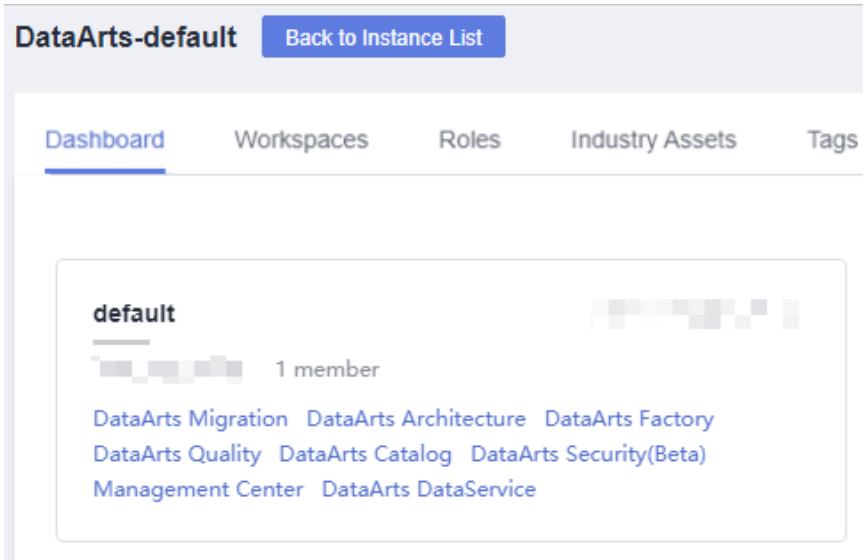
Parameter	Description
Blocked	**Blocked** is displayed when **Block all other accesses** is not selected. Users and user groups that are not allowed to access the system can be specified in the **Blocked** area. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are not allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Policy Conditions**: the range of IP addresses that can access the Kafka topic. • **Granted**: If **Granted** is selected, management permissions are assigned to appropriate users and groups. Delegated administrators can update and delete policies and create sub-policies based on the original policies.

----End

Creating a Storm Permission Policy

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-145 DataArts Security

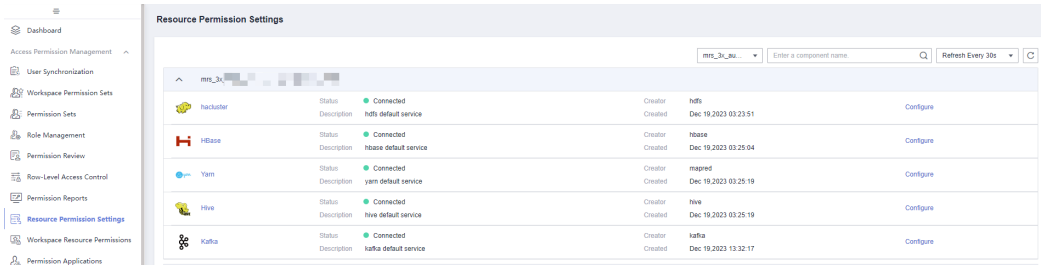


Step 2 Choose **Access Permission Management** > **Resource Permission Settings** from the left navigation bar.

NOTE

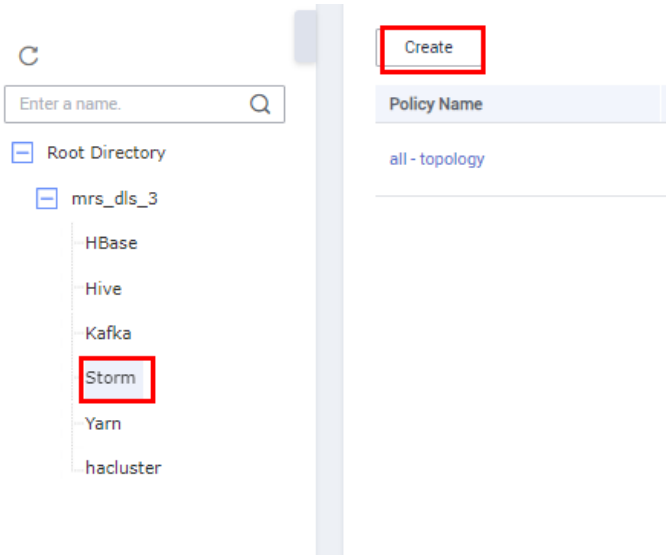
If error message "cluster [mrs_3x_autotest_do_not_del] get service failed. due to [null response from cdm:[404 NOT FOUND]]." is displayed, check whether the RangerAdmin service IP address and Ranger service port of the Ranger connection are correct in Management Center by referring to **Configuring an MRS Ranger Connection**.

Figure 9-146 Resource Permission Settings page



Step 3 Click **Configure** to the right of the Storm component, and click **Create** in the upper part of the page displayed.

Figure 9-147 Creating a Storm permission policy



Step 4 Set the parameters and click **OK**.

Figure 9-148 Configuring a Storm policy

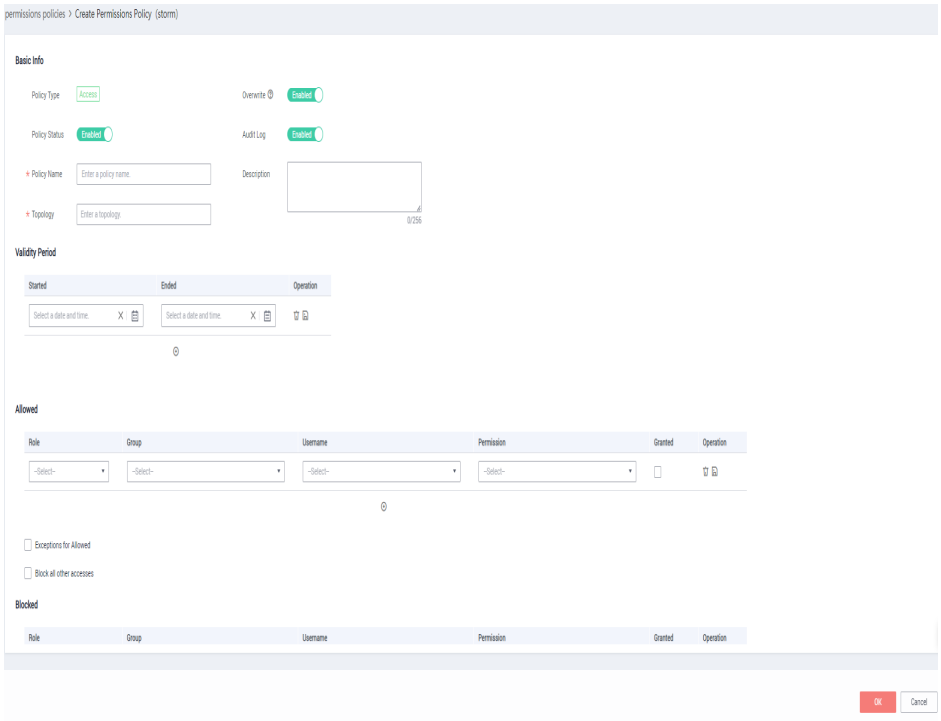


Table 9-25 Parameters of a Storm permission policy

Parameter	Description
Policy Type	The policy type is automatically generated based on the selected service component. **Policy Type** can be set to **Access**, **Mask**, and **Row-level Filter**. **Mask** and **Row-level Filter** are specific to Hive.
Policy Status	If **Policy Status** is **Enabled**, the permission policy takes effect immediately. If **Policy Status** is **Disabled**, the permission policy does not take effect after being created. **Policy Status** is set to **Enabled** by default.
Overwrite	If **Overwrite** is set to **Enabled**, the new policy takes effect and the old policy does not take effect. **Overwrite** is set to **Enabled** by default. To create a temporary access policy, enable **Overwrite** and set **Validity Period** as required. In this way, even if the temporary access policy expires, the original permission policy still takes effect.
Audit Log	If **Audit Log** is set to **Enabled**, logs are recorded. The log content includes the client access time, client IP address, client user, and resource operation result.
Policy Name	Policy name is mandatory. A policy name can include only letters, numbers, underscores (_), and hyphens (-). Up to 50 characters are allowed.

Parameter	Description
Description	A description of the policy. Up to 256 characters are allowed.
Topology	Tasks in a Storm cluster.
Validity Period	You can set the effective time and expiration time of a policy. You can configure multiple time ranges.
Allowed	Users and user groups that are allowed to access the resources. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Granted**: If **Granted** is selected, management permissions are assigned to appropriate users and groups. Delegated administrators can update and delete policies and create sub-policies based on the original policies.
Exceptions	If you select **Exceptions for Allowed**, users who are not allowed to access the system are added to the user group that is allowed to access the system. If you select **Exceptions for blocked**, users who are allowed to access the system are added to the user group that is blocked from the system.
Block all other accesses	If you select **Block all other accesses**, only specified users or user groups are allowed to access the system.
Blocked	**Blocked** is displayed when **Block all other accesses** is not selected. Users and user groups that are not allowed to access the system can be specified in the **Blocked** area. • **Username**: MRS user. • **Role**: MRS role. • **Group**: MRS user groups. • **Permission**: the permission required by users who are not allowed to access the system. **Permission** and **Username** can be left blank or not left blank at the same time. For details on service permissions, see **Table 9-17**. • **Granted**: If **Granted** is selected, management permissions are assigned to appropriate users and groups. Delegated administrators can update and delete policies and create sub-policies based on the original policies.

----End

9.3.7.2 Viewing Permission Reports

This section describes how to view the resource permission policies and policy details.

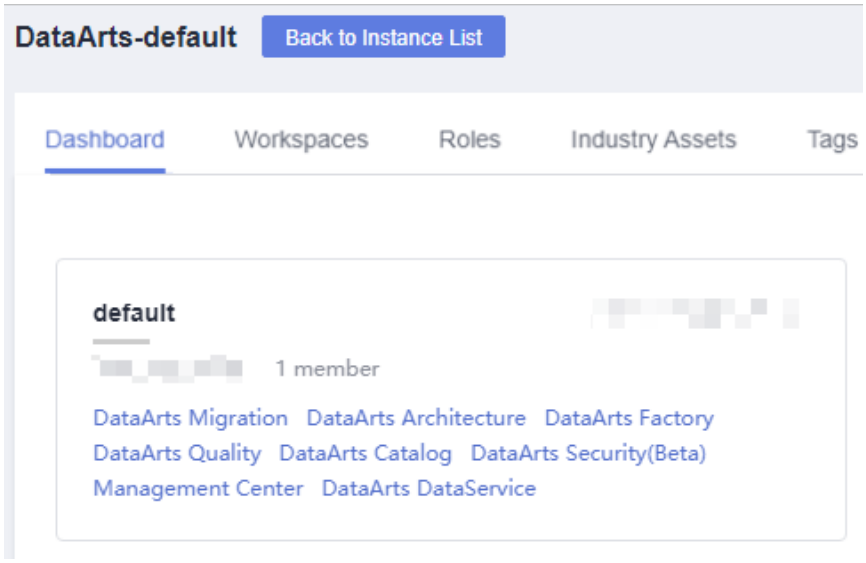
Prerequisites

The permission policy has been configured. For details on how to configure a permission policy, see **Configuring Resource Permissions**.

Viewing the Details of a Policy

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-149 DataArts Security



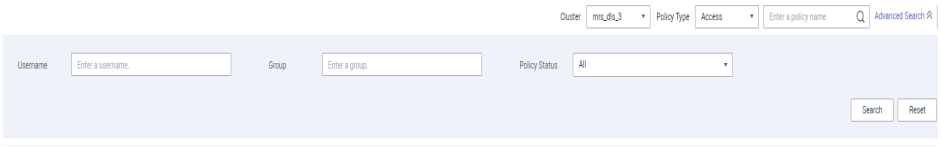
Step 2 Choose **Permission Reports** from the left navigation bar.

Step 3 Choose an MRS cluster (Ranger), and select a service to view its policies and policy details.

- Advanced search:

  When viewing a report, you can search for policies by cluster, policy name, username, user group, policy type, or policy status. You only need to click **Advanced Search** in the upper right corner of the **Permission Reports** page to display the search box.

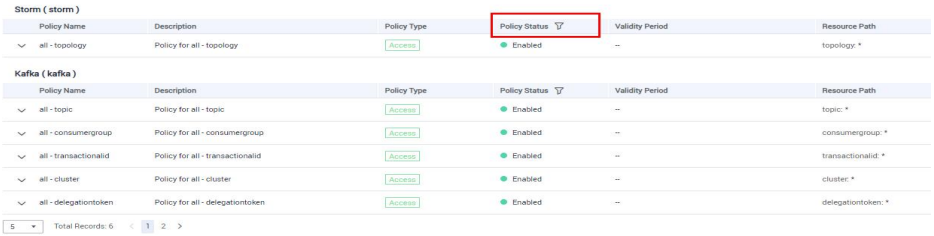
  Figure 9-150 Advanced search

- Policy status filtering:

In the policy list of a service, you can click in the **Policy Status** column to filter the policies to be viewed.

Figure 9-151 Policy status filtering



----End

9.4 Sensitive Data Governance

9.4.1 Sensitive Data Governance Process

Sensitive Data Definition

Sensitive data is usually used by others without the consent of individuals or companies. The interests of individuals or companies might be seriously compromised.

According to *GB/T 35273-2020 Information Security Technology — Personal Information Security Specification*, sensitive personal data includes:

- Personal property information (deposit, credit, and banking transactions)
- Personal health state and physiological information (physical examination information and medical records)
- Personal biometric information (fingerprint and facial features)
- Personal identity information (ID card, social security card, and driving license)
- Other information (religious belief and precise location)

Sensitive Data Protection Methods

- **Sensitive data identification and label adding**

  Classify and grade data to facilitate security management of different granularities and levels.
- **Data leakage detection and prevention**

  If sensitive data is frequently accessed, a risk alarm is reported immediately.
- **Static data masking and data watermarking**

  Sensitive data with a specific security level can be masked or watermarked when being provided to external systems.
- **Personal information compliance**

  Accurately distinguish and protect personal data to avoid compliance issues.

- **General data protection regulation (GDPR) compliance**

  Comply with GDPR requirements on detecting and protecting sensitive data, and audit the use of sensitive data.

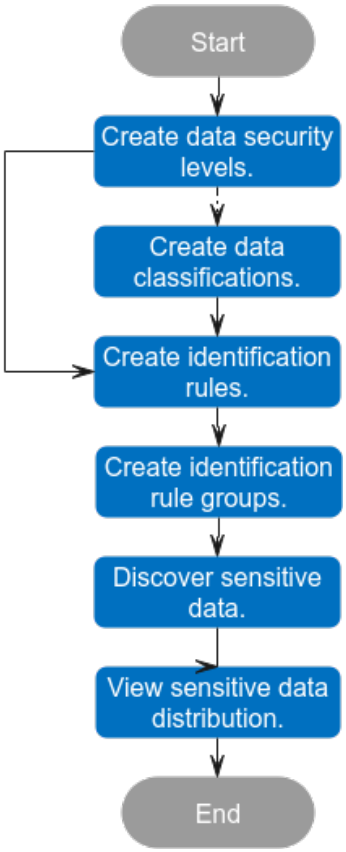
- **Data security compliance check**

  Based on the analysis of sensitive data, develop data security compliance management regulations to help enterprises build and improve their information security compliance management systems.

Sensitive Data Identification Process

Figure 9-152 shows the sensitive data identification process.

Figure 9-152 Sensitive data identification process



1. **Create data security levels.**

   Before performing any operations on data, create security levels for the data to specify the scope of confidential information.

2. **Create data classifications.**

   If data security levels cannot meet the data classification requirements in the case of a large amount of data, you can create data classifications for data of different values to better manage and measure your data.

3. **Create identification rules.**

   Define sensitive data identification standards.

4. **Create identification rule groups.**

Define sensitive data identification rules and rule groups for the purpose of effectively identifying sensitive data in a database.

5. **Discover sensitive data.**

   Create and run a sensitive data identification task.

6. **View sensitive data distribution.**

   View the sensitive data identified by the sensitive data identification task.

9.4.2 Creating Data Security Levels

To facilitate data management, you need to create data security levels and describe data confidentiality, for example, you can specify the application scope of your data. This section describes how to create data security levels.

Data security levels, data classifications, and identification rules are DataArts Studio instance-level configurations and can be exchanged between workspaces. In this way, data can be managed based on unified standards in the Data Map component.

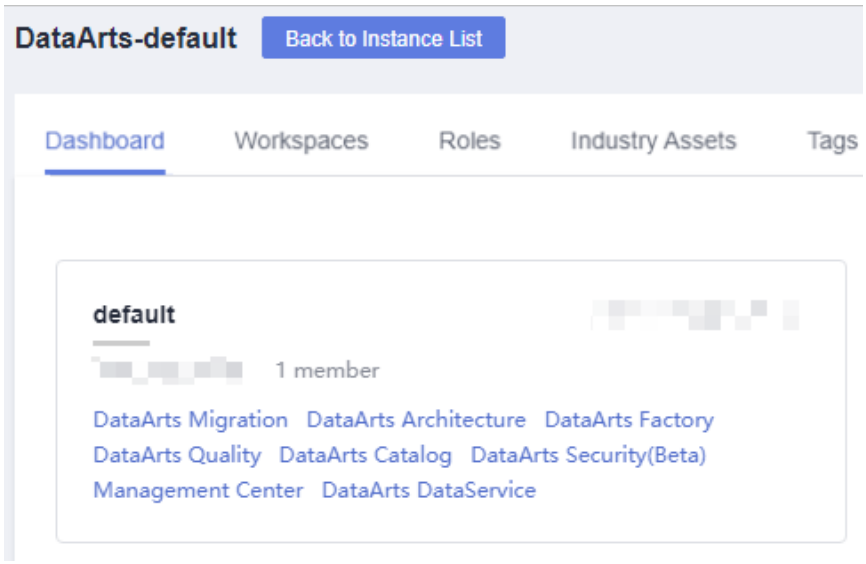
Constraints

- According to the industry common practice, a larger number indicates a higher security level. A maximum of 10 security levels can be created.
- Only the DAYU Administrator, Tenant Administrator, or data security administrator can create, modify, or delete data security levels, classifications, and identification rules. Other common users do not have permission to perform these operations.

Creating a Security Level

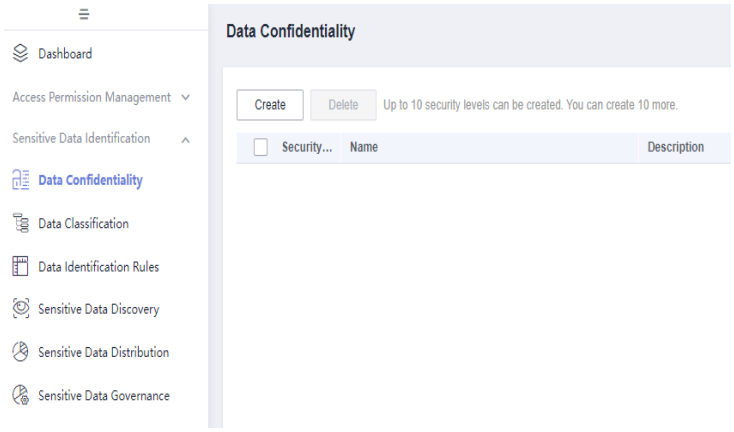
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-153 DataArts Security



Step 2 In the left navigation pane, choose **Data Confidentiality**.

Figure 9-154 Data Confidentiality page



Step 3 On the displayed page, click **Create** and set the parameters listed in **Table 9-26**.

Figure 9-155 Creating a data security level

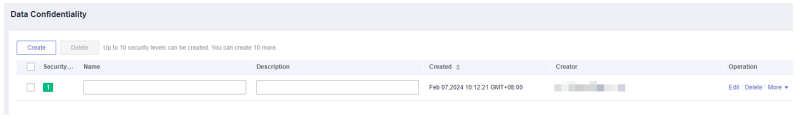


Table 9-26 Parameters

Parameter	Description
*Name	The security level name can include only letters, numbers, and underscores (_). After a security level is created, its name cannot be edited.
Description	All characters can be entered in a security level description. After a security level is created, you can edit its description.

NOTE

By default, security levels are displayed in ascending order. You can also move a security level up or down as required.

----End

Related Operations

- Adjusting a security level: On the **Data Confidentiality** page, locate a security level, click **More** in the **Operation** column, and select **Up** or **Down**.

- Editing a security level: On the **Data Confidentiality** page, locate a security level and click **Edit** in the **Operation** column to change the description of the security level.
- Deleting one or more security levels: On the **Data Confidentiality** page, locate a security level and click **Delete** in the **Operation** column to delete the security level. To delete multiple security levels, select them and click **Delete** above the list.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.

9.4.3 Creating Data Classifications

If data security levels cannot meet the data classification requirements in the case of a large amount of data, you can create data classifications for data of different values to better manage and measure your data. Data of different classifications are parallel, equal, and mutually exclusive. This section describes how to create data classifications.

Data security levels, data classifications, and identification rules are DataArts Studio instance-level configurations and can be exchanged between workspaces. In this way, data can be managed based on unified standards in the Data Map component.

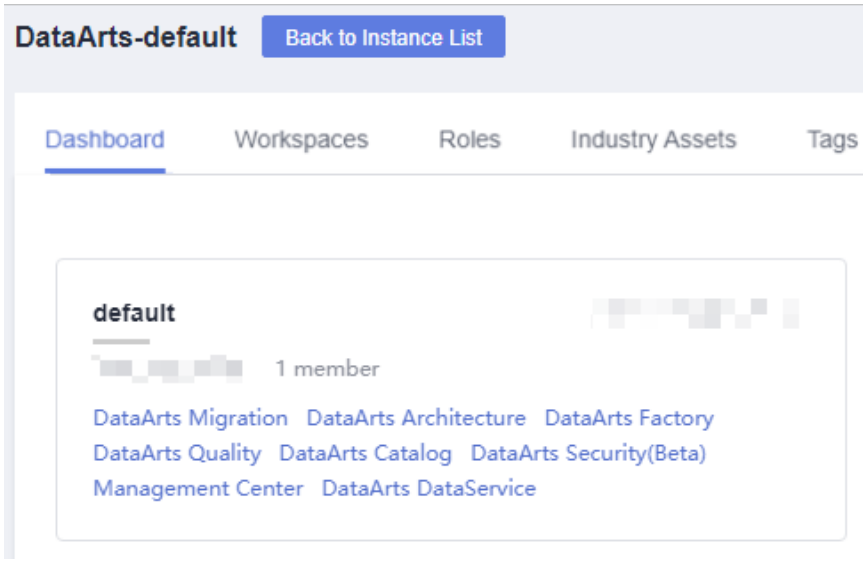
Constraints

- A maximum of 1,000 data classifications at five layers are allowed.
- Only the DAYU Administrator, Tenant Administrator, or data security administrator can create, modify, or delete data security levels, classifications, and identification rules. Other common users do not have permission to perform these operations.
- If a parent classification contains sub-classifications, the parent classification can be deleted only after the sub-classifications have been deleted.

Creating a Classification

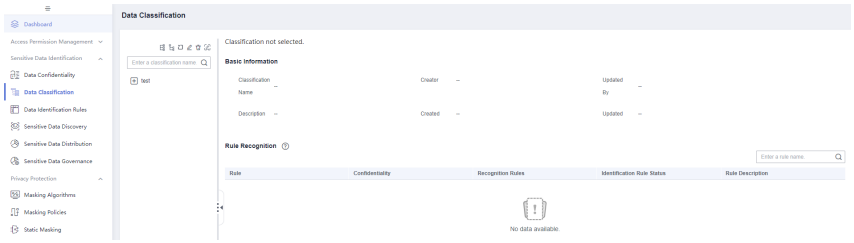
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-156 DataArts Security



Step 2 In the left navigation pane, choose **Data Classification**.

Figure 9-157 Data Classification page



Step 3 Before creating your first classification, click above the classification directory to add at least one root classification. Then you can click or to add a classification of the same level or a sub-classification.

After you click or, set parameters in the displayed dialog box by referring to **Table 9-27**.

Figure 9-158 Creating a data classification

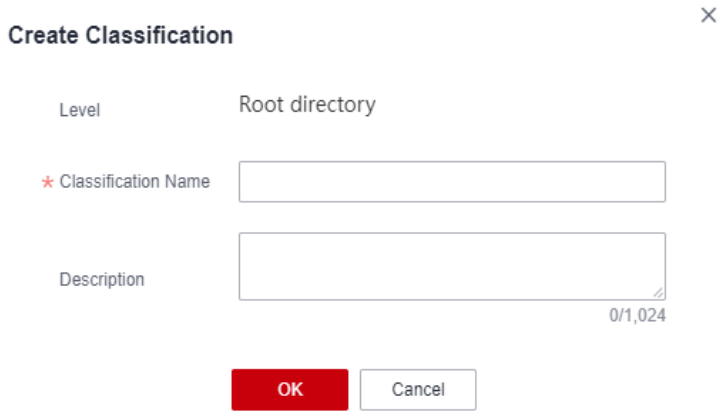


Table 9-27 Parameters

Parameter	Description
*Classification Name	The classification name can contain only letters, digits, and underscores (_). After the classification is created, it cannot be edited.
Description	The description supports all types of characters and can be edited.

----End

Related Operations

- Editing a classification: On the **Data Classification** page, select the classification to be modified and click above the classification directory to change the classification name or description.
- Deleting a classification: On the **Data Classification** page, select the classification to be deleted and click above the classification directory to delete the classification.

  You can also delete classifications by editing the data classification directory. To be specific, you can click above the classification directory and delete classifications on the displayed **Edit Data Classification Directory** page.

  NOTE

  - If a parent classification contains sub-classifications, the parent classification can be deleted only after the sub-classifications have been deleted.
  - The deletion operation cannot be undone. Exercise caution when performing this operation.
- Editing a classification directory: Click above the classification directory. On the **Edit Data Classification Directory** page, you can add sub-classifications or delete classifications.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.

9.4.4 Creating Identification Rules

To effectively identify sensitive data fields in a database, you can create identification rules.

Data security levels, data classifications, and identification rules are DataArts Studio instance-level configurations and can be exchanged between workspaces. In this way, data can be managed based on unified standards in the Data Map component.

NOTE

After an identification rule is created, it remains to be confirmed by default and cannot take effect for a static masking task. To make the identification rule take effect, perform the following operations:

After running a sensitive data discovery task, you must choose **Sensitive Data Distribution** in the left navigation pane, click the **Manual Recovery** tab, and ensure that the identification rule of the task is valid, so that the rule can take effect for dynamic masking tasks.

Prerequisites

- (Mandatory) A data security level has been created. For details, see **Creating Data Security Levels**.
- (Optional) A data classification has been created. For details, see **Creating Data Classifications**.

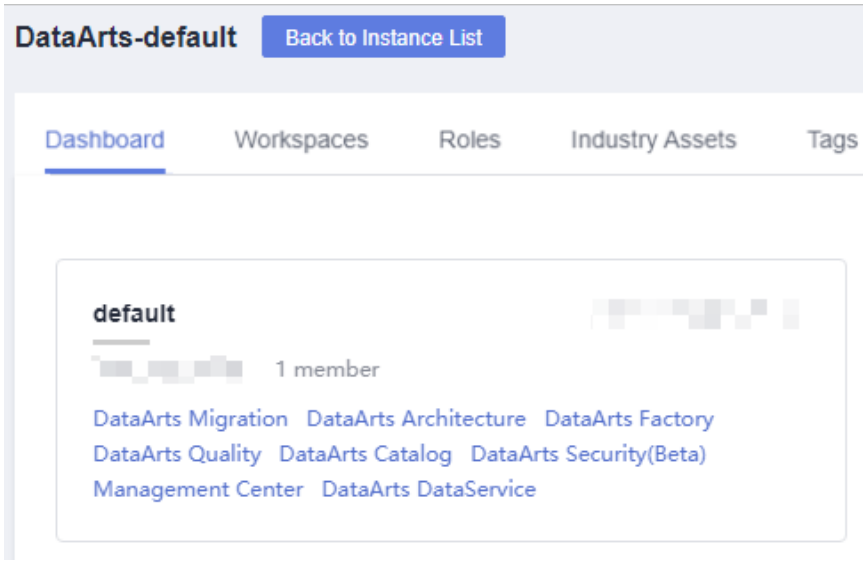
Constraints

- Only the DAYU Administrator, Tenant Administrator, or data security administrator can create, modify, or delete data security levels, classifications, and identification rules. Other common users do not have permission to perform these operations.
- If the sensitive data identification rule is of the content identification type (that is, a built-in rule or a custom rule of the content identification type), a field is considered as a sensitive field and matched with a security level and classification only when the proportion of the number of records that match the identification rule of a field to the total number of records in the data table exceeds a specified threshold (80% by default).

Creating a Data Identification Rule

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

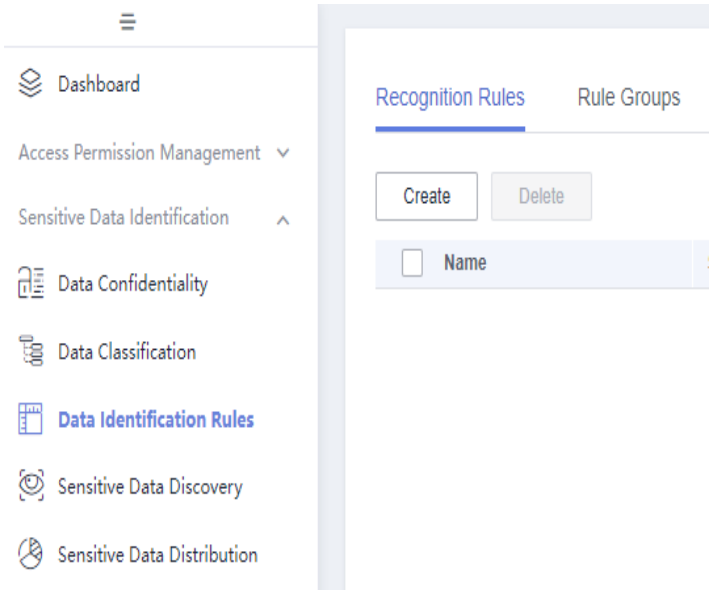
Figure 9-159 DataArts Security



Step 2 In the left navigation pane, choose **Data Identification Rules**.

Step 3 On the displayed page, click **Create**.

Figure 9-160 Creating a data identification rule



Step 4 Set the parameters based on **Table 9-28** and click **OK**.

Figure 9-161 Setting parameters for the rule

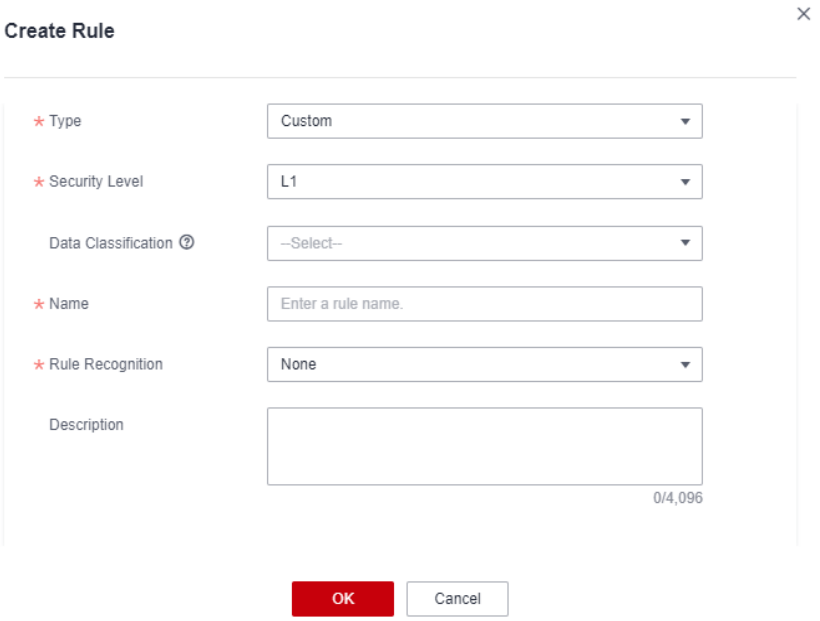


Table 9-28 Parameters

Parameter	Description
*Type	The category to which a rule belongs. You can either create a rule based on built-in templates or customize one.
*Security Level	Classify the configured data into different levels. If the existing security levels do not meet the requirements, go to the **Data Confidentiality** page to create security levels. For details, see **Creating Data Security Levels**.
Data Classification	Classify the configured data into different types. If the existing classifications do not meet the requirements, go to the **Data Classification** page to create classifications. For details, see **Creating Data Classifications**.
Description	A description of the rule to be created.
Built-in	
*Template	This parameter is displayed when **Type** is set to **Built-in**. The system provides more than 70 preset sensitive data identification and masking rules for the following information: sensitive personal information (such as phone numbers, debit cards, and credit cards), sensitive enterprise information (such as financial asset information and delivery information), sensitive key information (such as DSA keys and RSA keys), sensitive device information (such as IPv4 and IPv6 addresses), sensitive location information (such as provinces, cities, GPS locations, and addresses), and general sensitive information (date).
*Name	If **Type** is set to **Built-in**, the rule name is automatically generated based on the template.
Custom	
*Name	If **Type** is set to **Custom**, you can enter a rule name, which is mandatory. You are advised to include the rule meaning into the rule name and avoid meaningless descriptions so that the rule can be quickly located and selected. **NOTE** The name must be unique.
*Rule Recognition	This parameter is displayed when **Type** is set to **Custom**. The options are **None** and **Regular**. If you select **None**, the sensitive data identification task associated with the rule does not take effect. Data assets cannot be automatically classified. You need to manually add categories.

Parameter	Description
*Regular	This parameter is displayed when **Regular** is set for **Rule Recognition**. • If you select **Content recognition**, enter a custom regular expression. The expression will be used to identify data content. Example: **^ male$|^female&**. • If you select **Column name recognition**, enter a custom regular expression. The expression will be used to accurately or fuzzily identify column names. Multiple column names can be identified at the same time. Example: **sex|gender**. • If you select **Remarks recognition**, enter a custom regular expression. The expression will be used to fuzzily identify remarks. Example: **.*comment.***.

----End

Related Operations

- Editing an identification rule: On the **Data Identification Rules** page, locate an identification rule and click **Edit** in the **Operation** column to change the security level, classification, and description of the identification rule. For a custom rule, you can also change the rule recognition and regular expression.
- Editing the identification rule status: The identification rule is enabled by default. If the identification rule is disabled, it cannot be added to an identification rule group.

  To change the status of the identification rule, click  or  to enable or disable the rule.
- Deleting identification rules: On the **Data Identification Rules** page, locate an identification rule and click **Delete** in the **Operation** column. To delete identification rules in a batch, select them and click **Delete** above the list.

  Note: Identification rules that have been referenced by identification rule groups or masking policies cannot be deleted. To delete such rules, modify the reference relationship first.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.
- Testing preset rule templates: On the **Preset Rule Templates** tab page, you can view all preset rule templates and test the recognition result of the templates by entering custom sample data.

9.4.5 Creating Identification Rule Groups

A sensitive data identification rule group has service logic and contains scattered rules. A rule group is the prerequisite for running a sensitive data discovery task.

Prerequisites

Identification rules have been created. For details, see **Creating Identification Rules**.

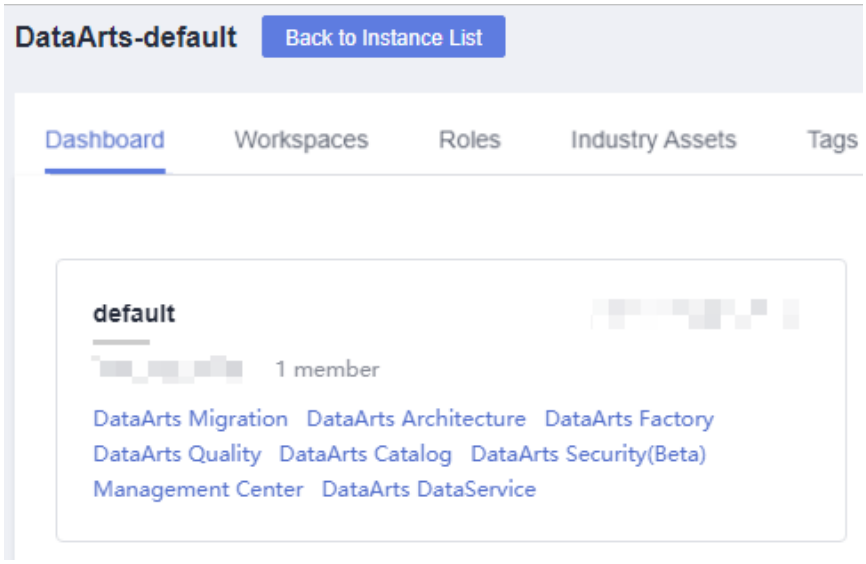
Constraints

- During sensitive data identification, if a field matches multiple identification rules in an identification rule group, the highest security level of the identification rules is used as the security level of the field, and multiple field classifications are allowed.
- A maximum of 100 identification rule groups can be created.

Creating a Sensitive Data Identification Rule Group

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

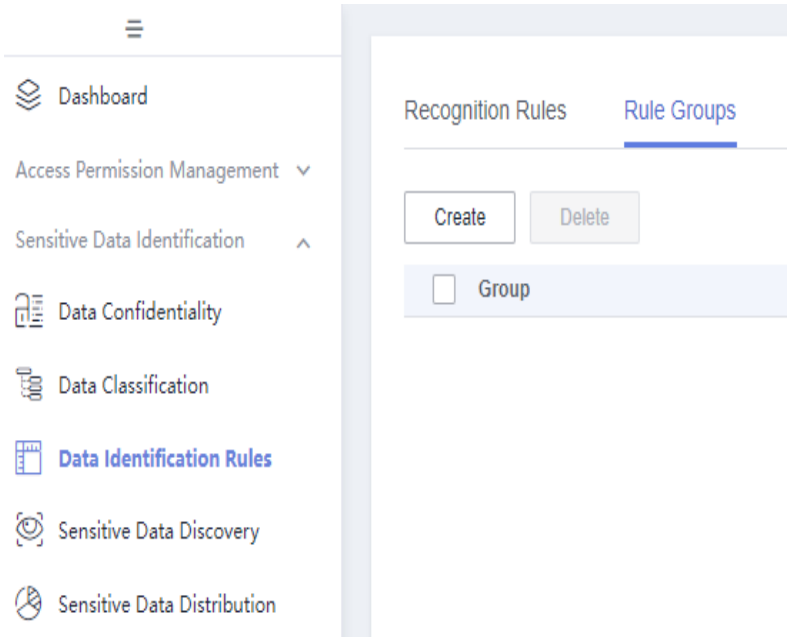
Figure 9-162 DataArts Security



Step 2 Choose **Data Identification Rules** from the left navigation bar.

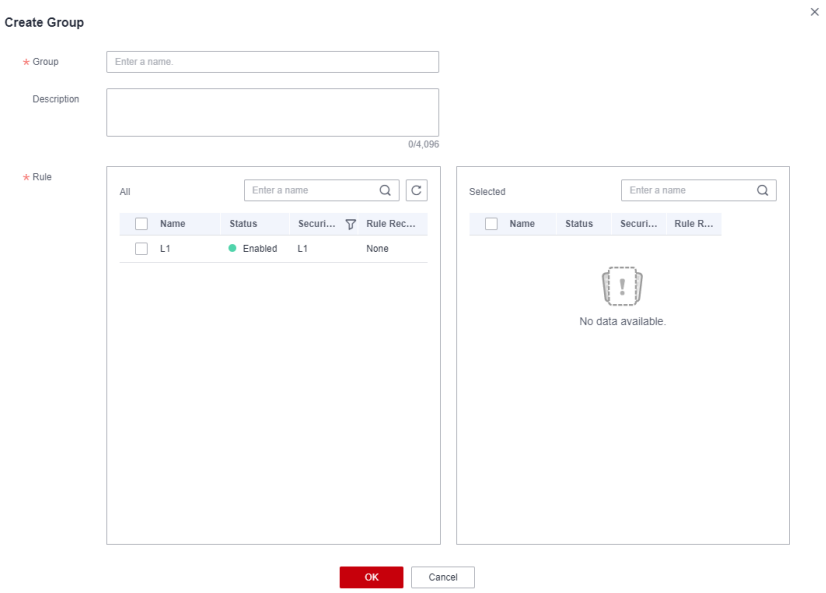
Step 3 Click the **Rule Groups** tab in the upper part of the displayed page.

Figure 9-163 Creating a sensitive data identification rule group



Step 4 Click **Create**, set the group name and description based on **Table 9-29**, select identification rules, and click **OK**.

Figure 9-164 Parameters for creating an identification rule group



The selected rules are displayed in the list on the right. You can click to deselect the selected rules.

Table 9-29 Parameters

Parameter	Description
*Group	Group names can include only letters, numbers, and underscores (_). You are advised to include the rule group meaning into the name and avoid meaningless descriptions so that the rule group can be quickly located and selected.
Description	Information to better identify the group

----End

Related Operations

- Editing a rule group: On the **Rule Groups** page, locate a group and click **Edit** in the **Operation** column to change the name, description, and rules of the group.
- Deleting a rule group: On the **Rule Groups** page, locate a group and click **Delete** in the **Operation** column. To delete rule groups in a batch, select them and click **Delete** above the list.

  A rule group that is being referenced cannot be deleted directly. To delete it, you must disassociate it from the sensitive data discovery task first by changing the **Identification Rule Group** parameter of the sensitive data discovery task based on **Discovering Sensitive Data**.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.

9.4.6 Discovering Sensitive Data

After creating a sensitive data identification rule group, you can create a sensitive data discovery task to discover sensitive data and synchronize it to Data Map.

NOTE

After running a sensitive data discovery task, you must choose **Sensitive Data Distribution** in the left navigation pane, click the **Manual Recovery** tab, and ensure that the identification rule of the task is valid, so that the rule can take effect for dynamic masking tasks.

Prerequisites

- Sensitive data identification rule groups have been created. For details, see **Creating Identification Rule Groups**.
- A DWS connection, a DLI connection, and an MRS Hive connection have been created in Management Center based on **Creating a Data Connection**.
- Before discovering DLI sensitive data, you must prepare a general-purpose DLI queue.
- To enable automatic synchronization of identified sensitive data to the Data Map component, the sensitive data discovery task must be created, run, or

scheduled by DAYU Administrator, Tenant Administrator, or data security administrator.

- To enable the synchronization of sensitive data classifications to the Data Map component, ensure that the following prerequisites are met:
  - You have collected the metadata of the data table in DataArts Catalog. For details, see **Metadata Collection Task**.
  - Real-time metadata synchronization has been enabled for the data connections in Management Center. For details, see **Creating a Data Connection**.

Constraints

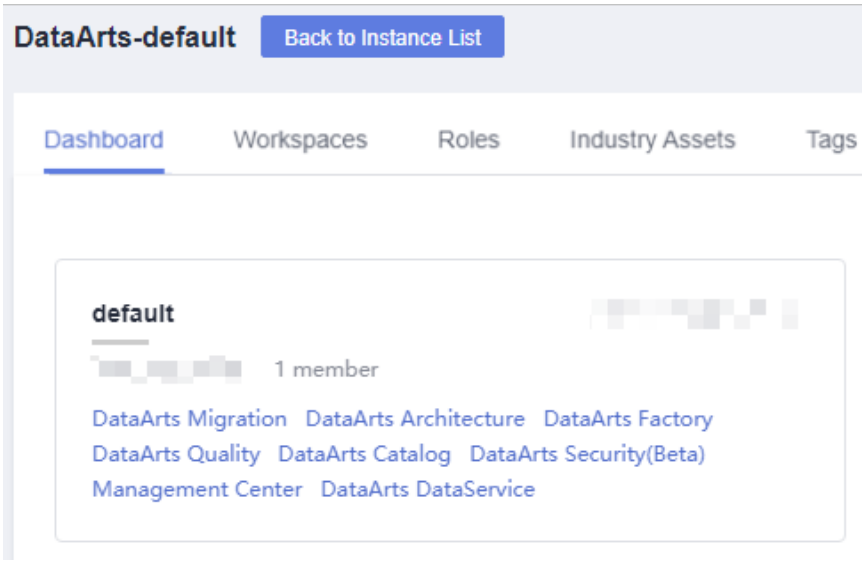
- Sensitive data discovery is only available for standard warehouses of GaussDB(DWS), Data Lake Insight (DLI), and MRS Hive.
- If the sensitive data identification rule is of the content identification type (that is, a built-in rule or a custom rule of the content identification type), a field is considered as a sensitive field and matched with a security level and classification only when the proportion of the number of records that match the identification rule of a field to the total number of records in the data table exceeds a specified threshold (80% by default).
- During sensitive data identification, if a field matches multiple identification rules in an identification rule group, the highest security level of the identification rules is used as the security level of the field, and multiple field classifications are allowed.
- After a sensitive data discovery task is executed, the security levels and classifications are generated for the discovered sensitive fields. By default, security levels of data tables are not generated. Security levels of data tables are generated only if you select **Update the security level**. The security level of a data table is the highest security level of sensitive fields.
- Currently, sensitive data can be synchronized only to Data Map. Sensitive data cannot be synchronized to DataArts Catalog, and sensitive data security levels and classifications cannot be added or edited in DataArts Catalog.
- Only the DAYU Administrator, Tenant Administrator, or data security administrator has the permission to enable automatic synchronization of sensitive data to Data Map or manually synchronize sensitive data to Data Map.
  - Automatic synchronization: If **Manually synchronize the recognition result** is not selected during the creation of a sensitive data discovery task, sensitive data is automatically synchronized to Data Map.
  - Manual synchronization: If you select **Manually synchronize the recognition result** when creating a sensitive data discovery task, you need to choose **Sensitive Data Distribution** and click the **Manual Recovery** tab to synchronize sensitive data to Data Map.

  When creating a sensitive data discovery task as a common user other than the DAYU Administrator, Tenant Administrator, or data security administrator, you must select **Manually synchronize the recognition result** so that the task can be successfully created. In addition, if you run or schedule a task for which **Manually synchronize the recognition result** is not selected as a common user, the task cannot be executed.

Creating a Sensitive Data Discovery Task

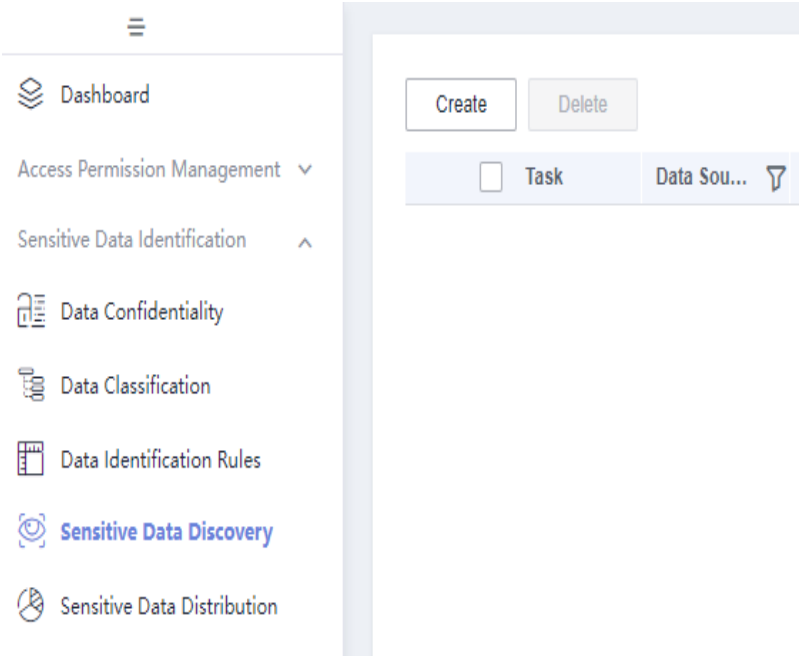
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-165 DataArts Security



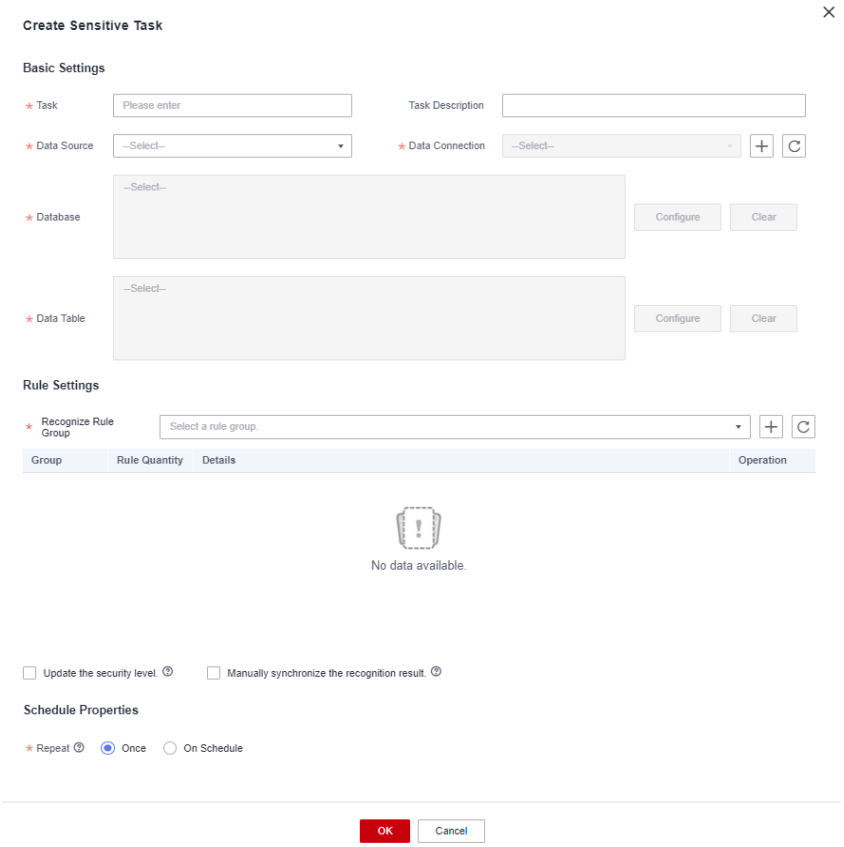
Step 2 Choose **Sensitive Data Discovery** from the left navigation bar.

Figure 9-166 Sensitive Data Discovery page



Step 3 Click **Create**. In the **Create Sensitive Task** slide-out panel, set parameters based on **Table 9-30**.

Figure 9-167 Setting parameters for the sensitive data discovery task



The following table lists the parameters for a sensitive data discovery task.

Table 9-30 Parameters

Parameter	Description
Basic Settings	
*Task	Name of the task. To facilitate task management, you are advised to include the data table to be identified and the rule group to be used in the task name.
Task Description	A description of the task to be created.
*Data Source	Select a created data source from the drop-down list.
*Data Connection	Select a data connection from the drop-down list. If no data connection is available, create one by referring to **Creating a Data Connection**.

Parameter	Description
*Database *Data Table	Databases and data tables where you want to discover sensitive data. • Click **Configure** following the **Database** box to select databases. • Click **Configure** following the **Data Table** box to select data tables. • Click **Clear** to delete the selected databases and data tables.
*Computing Queue	This parameter is mandatory if **Data Source** is set to **DLI**. Select a general-purpose DLI queue for executing DLI jobs.
Rule Settings	
*Recognize Rule Group	Select a rule group from the drop-down list. If no rule groups are created, create one by referring to **Creating Identification Rule Groups**. When you select a group, details about the identification rules in the group are displayed. You can configure thresholds for preset rules and custom rules that contain content matching. When the proportion of the number of records that match the identification rule of a field to the total number of records in the data table exceeds the threshold (80% by default), the field is considered sensitive. If different rule groups contain the same rule, the threshold for the rule must be the same.
Update the security level	After the sensitive data discovery task is executed, the security levels and classifications are generated for the identified sensitive fields. By default, this option is not selected, indicating that the security levels of data tables are not generated. If this option is selected, the security levels of data tables are generated. The security level of a data table is the highest security level of the sensitive fields.

Parameter	Description
Manually synchronize the recognition result	Only the DAYU Administrator, Tenant Administrator, or data security administrator has the permission to enable automatic synchronization of sensitive data to Data Map or manually synchronize sensitive data to Data Map. • Automatic synchronization: If **Manually synchronize the recognition result** is not selected during the creation of a sensitive data discovery task, sensitive data is automatically synchronized to Data Map. • Manual synchronization: If you select **Manually synchronize the recognition result** when creating a sensitive data discovery task, you need to choose **Sensitive Data Distribution** and click the **Manual Recovery** tab to synchronize sensitive data to Data Map. When creating a sensitive data discovery task as a common user other than the DAYU Administrator, Tenant Administrator, or data security administrator, you must select **Manually synchronize the recognition result** so that the task can be successfully created. In addition, if you run or schedule a task for which **Manually synchronize the recognition result** is not selected as a common user, the task cannot be executed.
Schedule Properties	
Once	The sensitive data discovery task runs only once.
On Schedule	The sensitive data discovery task runs based on the configured scheduling period. • **Date** Period during which the task takes effect • **Cycle** The frequency at which a task is executed. The options are: – **minutes**: Select the scheduling start time and end time, and set the interval in minutes. – **hours**: Select the scheduling start time and end time, and set the interval in hours. – **Day**: Set the scheduling time everyday. – **Week**: Select a day in a week and set the specific time to start scheduling. – **Month**: Select a day in a month and set the specific time to start scheduling. For example, you can set **Cycle** to **Week**, **Time** to **15:52**, and **Time Range** to **Tuesday**. In this case, the task is executed at 15:52 every Tuesday within the configured date range. • **Start now**: If you select this option, the task is scheduled immediately.

Step 4 Click **OK**. The sensitive data discovery task is created.

NOTE

If no execution result is displayed after the sensitive data discovery task is successfully executed, and no matched information is found in the run log, it means no sensitive data is discovered.

----End

Related Operations

- Running or scheduling a task: On the **Sensitive Data Discovery** page, locate a task and click **Run** in the **Operation** column or click **More** in the **Operation** column and select **Start**.

  You can determine whether a task is scheduled once or repeatedly based on the scheduling period.

  NOTE

  If you run or schedule a task for which **Manually synchronize the recognition result** is not selected as a common user other than the DAYU Administrator, Tenant Administrator, or data security administrator, the task fails to be executed. Only the DAYU Administrator, Tenant Administrator, or data security administrator can run or schedule tasks for which **Manually synchronize the recognition result** is not selected.

- Editing a task: On the **Sensitive Data Discovery** page, locate a task and click **Edit** in the **Operation** column.

  A task in the **Running** state cannot be edited.

- Deleting tasks: On the **Sensitive Data Discovery** page, locate a task, click **More** in the **Operation** column, and select **Delete**. To delete multiple tasks at a time, select the tasks and click **Delete** above the task list.

  A task in the **Running** state cannot be deleted.

  NOTE

  - Deleting a sensitive data discovery task will delete the discovery result. Exercise caution when performing this operation.
  - The deletion operation cannot be undone. Exercise caution when performing this operation.

- Viewing running instance logs: On the **Sensitive Data Discovery** page, locate a task and click to expand instances. Click **Operation** and select **View Log**.

  If a task fails to be executed, you can locate the failure cause based on logs, rectify the fault, and try the task again. If the fault persists, contact technical support.

9.4.7 Viewing Sensitive Data Distribution

This section describes how to view and modify the sensitive data discovery result.

- View the result of a sensitive data discovery task.
- Manual recovery: After sensitive data is discovered, you must perform manual recovery to confirm that the identification rule in the task is in valid state so that the identification rule takes effect for static masking tasks.

If you selected **Manually synchronize the recognition result** for the sensitive data discovery task, you must click **Data Sync** to synchronize the discovered sensitive data to Data Map. (Before synchronizing data, ensure that a metadata collection task has been performed in DataArts Architecture. Otherwise, the synchronization will fail.)

Prerequisites

- You have created and executed a sensitive data discovery task. For details, see **Creating a Sensitive Data Discovery Task**.
- Only the DAYU Administrator, Tenant Administrator, or data security administrator has the permission to synchronize sensitive data to Data Map.
- Before synchronizing sensitive data, you have collected the metadata of the data connection in DataArts Catalog. For details, see **Metadata Collection Task**. Otherwise, the synchronization will fail and an error message will be displayed, indicating that no data connection is available.

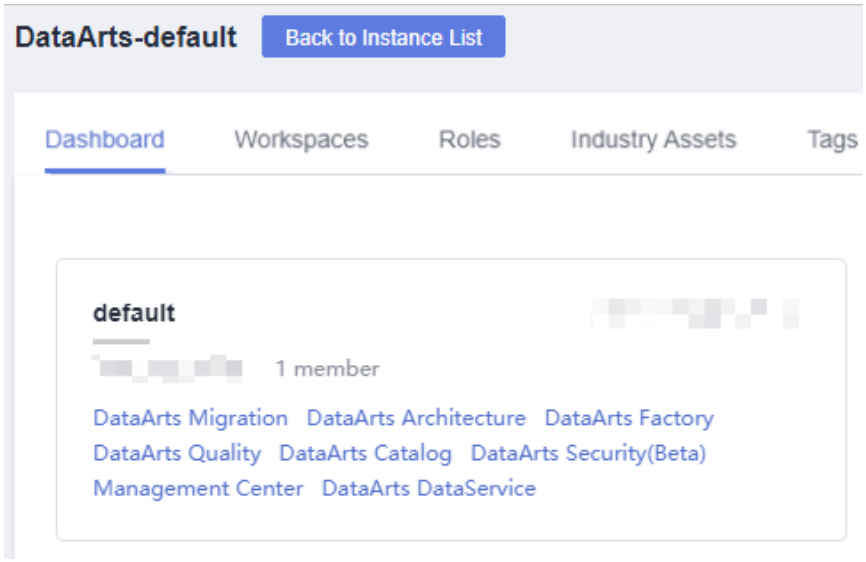
Constraints

- Currently, sensitive data can be synchronized only to Data Map. Sensitive data cannot be synchronized to DataArts Catalog, and sensitive data security levels and classifications cannot be added or edited in DataArts Catalog.
- Sensitive data synchronization depends on metadata collection tasks. If the metadata of a data connection has not been collected, no data connection can be found.

Viewing and Modifying the Sensitive Data Discovery Result

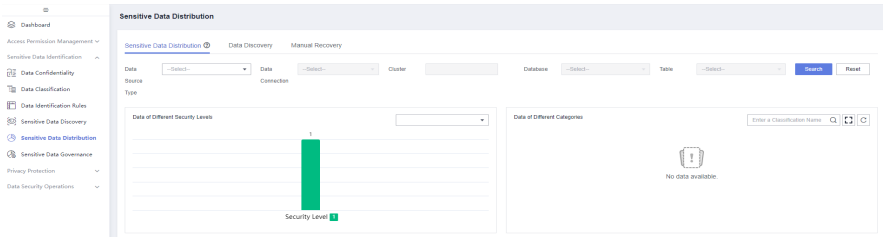
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-168 DataArts Security



Step 2 In the navigation pane on the left, choose **Sensitive Data Distribution**.

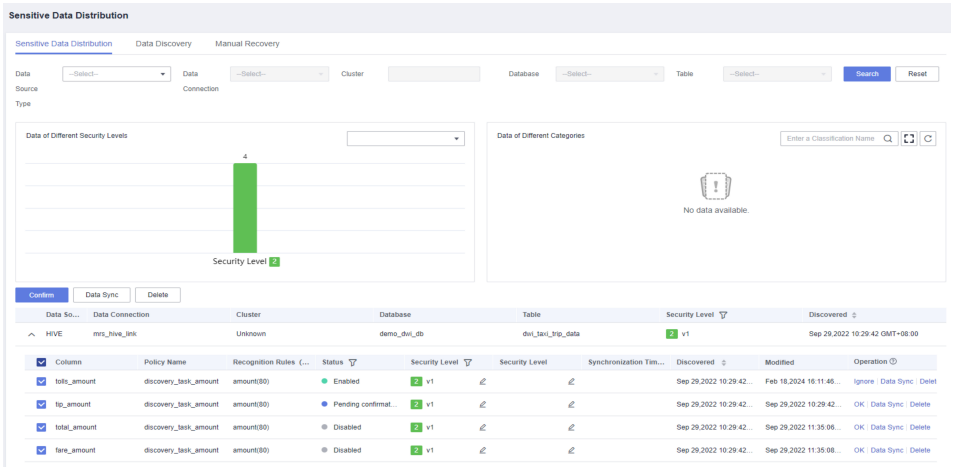
Figure 9-169 Sensitive Data Distribution page



Step 3 On the **Sensitive Data Distribution** page, you can use either of the following methods to view and modify the sensitive data discovery result. Method 1 is recommended. It allows you to view and modify the discovered sensitive data, change the data security level and classification, and perform batch operations without switching to other pages.

1. (Recommended) On the **Sensitive Data Distribution** tab page, click to expand data source details, view sensitive data, and change the data security level, classification, and status.
   - **OK**: Confirm that the identification result is valid. You can confirm rules in unconfirmed or invalid state. Static masking tasks can be executed using valid identification rules.
   - **Ignore**: Confirm that the identification result is invalid. You can ignore rules in valid state. Unconfirmed or invalid identification rules cannot be selected for static masking tasks.
   - **Data Sync**: If you selected **Manually synchronize the recognition result** for the sensitive data discovery task, you must click **Data Sync** to synchronize the discovered sensitive data to Data Map. (Before synchronizing data, ensure that a metadata collection task has been performed in DataArts Architecture. Otherwise, the synchronization will fail.)
   - **Delete**: Delete the discovered result.

   Figure 9-170 Viewing and modifying the discovered sensitive data

2. Click the **Data Discovery** tab. Search for a data source and click **View** in the **Operation** column to view details of the sensitive data.

Figure 9-171 Data discovery

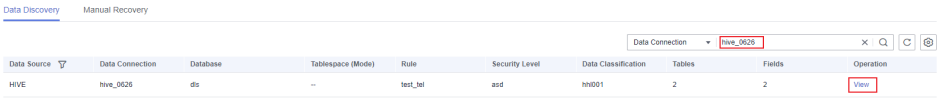


Figure 9-172 Viewing details

Click the **Manual Recovery** tab, search for and locate a rule, and click **OK**, **Ignore**, or **Data Sync** in the **Operation** column to change the data status.

- **OK**: Confirm that the identification result is valid. You can confirm rules in unconfirmed or invalid state. Static masking tasks can be executed using valid identification rules.
- **Ignore**: Confirm that the identification result is invalid. You can ignore rules in valid state. Unconfirmed or invalid identification rules cannot be selected for static masking tasks.
- **Data Sync**: If you selected **Manually synchronize the recognition result** for the sensitive data discovery task, you must click **Data Sync** to synchronize the discovered sensitive data to Data Map. (Before synchronizing data, ensure that a metadata collection task has been performed in DataArts Architecture. Otherwise, the synchronization will fail.)

Figure 9-173 Modifying sensitive data

----End

9.4.8 Managing Sensitive Data

With DataArts Security, you can manage Data Map assets by security level and control users' access to metadata. After you configure a security level for a specified user or user group, the user or user group can only access assets whose security levels are lower than or equal to the configured security level.

The security level-based permission control policies configured for a DataArts Studio instance are visible to and take effect for all the workspaces of the

instance. If no security level-based permission control policy is configured, DataArts Security provides a default policy. This policy grants the permission to access data of the highest security level to all users by default. After the administrator configures a policy, the default policy can be deleted.

Prerequisites

- A sensitive data discovery task has been performed and discovered sensitive data has been automatically or manually synchronized to Data Map. For details, see **Discovering Sensitive Data** or **Viewing Sensitive Data Distribution**.

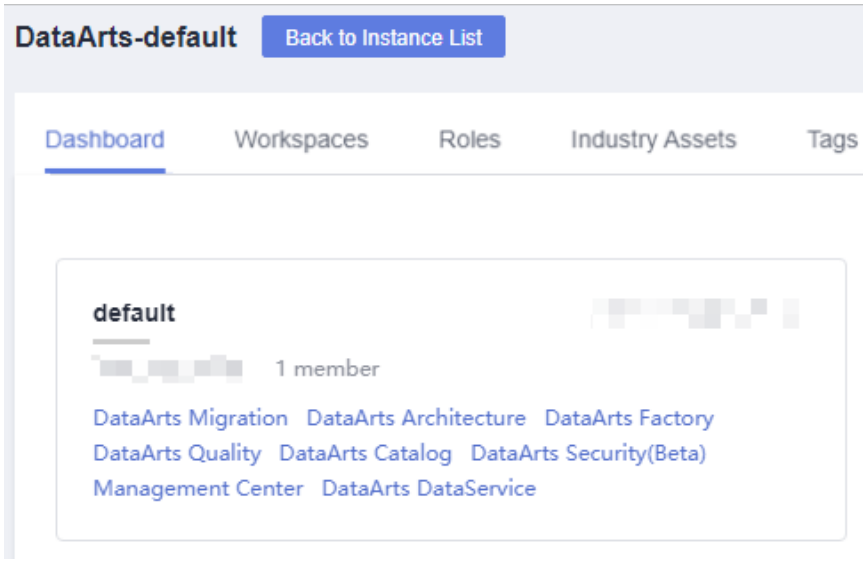
Constraints

- Only the DAYU Administrator, Tenant Administrator, or data security administrator can create, modify, or delete security level-based permission control policies. Other common users do not have permission to perform these operations.
- Security level-based permission control is available only for the fields with security levels in Data Map and unavailable for tables with security levels.
- A user/user group and a security level uniquely identify a security level-based permission control policy. A policy for the same user, user group, or security level cannot be created.
- If a user or user group corresponds to multiple security levels, the highest security level prevails.

Creating a Sensitive Data Control Policy

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

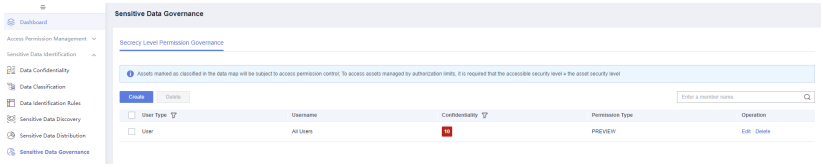
Figure 9-174 DataArts Security



Step 2 In the navigation pane on the left, choose **Sensitive Data Governance**.

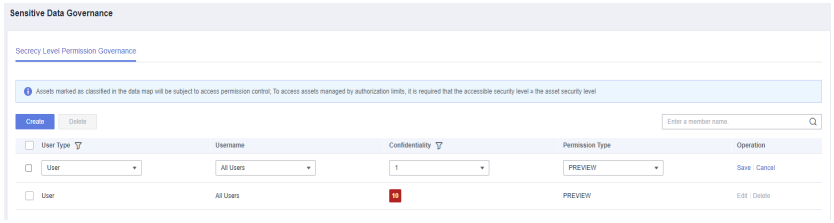
A default policy is displayed on the page. This policy grants all users the permission to access data with the highest security level.

Figure 9-175 Sensitive Data Governance page



Step 3 Click **Create** and set the parameters listed in **Table 9-31**.

Figure 9-176 Setting parameters for a security level-based permission control policy



The following table lists the parameters for the security level-based permission control policy.

Table 9-31 Policy parameters

Parameter	Description
*User Type	Select **User** or **User Group**.
*Username	Select a user or user group from all workspace members of the current instance.
*Confidentiality	Select a security level for the specified user or user group. The user or user group can only access assets whose security levels are lower than or equal to the configured security level.
*Permission Type	Only **PREVIEW** in Data Map is available.

Step 4 Click **Save**. After creating the policy, delete the default policy to make the created policy take effect.

----End

Related Operations

- Editing a security level-based permission control policy: On the **Sensitive Data Governance** page, locate a policy and click **Edit** in the **Operation** column to change the user/user group, confidentiality, or permission type.

- Deleting security level-based permission control policies: On the **Sensitive Data Governance** page, locate a policy and click **Delete** in the **Operation** column to delete the policy. To delete multiple policies, select them and click **Delete** above the policy list.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.

9.5 Privacy Protection and Management

9.5.1 Overview

DataArts Security provides privacy data protection to protect enterprises' sensitive data. You can use static and dynamic data masking, and data, file, and dynamic watermarking to prevent your data from being misused, disclosed, or stolen intentionally or unintentionally. In this way, your sensitive data is secure, complete, and safe to use.

Methods

Privacy data protection provides the following methods for protecting sensitive data:

- Static masking

  Static data masking prevents private data leakage, and ensures regulatory compliance as well as data security for enterprises. Sensitive data is masked, truncated, and hashed based on the abundant and effective built-in masking algorithms, and the processed data can be written to the target data table. For security purpose, it is the target data table that can be used to provide services for external requirements.

- Dynamic data masking

  After a dynamic masking policy is created in DataArts Security, the system synchronizes the policy to the data source. The data source dynamically masks data columns based on specified rules. When the users and user groups specified in the policy access sensitive data, the system returns the data that is dynamically masked by the data source to protect sensitive data from being disclosed.

- Data watermarking

  Users can embed watermarks into data. The watermarked data is transparent, available, and covert. It is not easy for others to crack the watermarked data. Even if data is leaked, watermarks can be traced to find the person accountable for the leakage. Once the watermarked data is used without the content of data owners, users can import the leaked file to trace and extract the watermarks. In this case, the organization or person that is accountable for the leakage problem can be easily found.

- File watermarks

  File watermarks can be injected into data files in the following scenarios to accurately locate security events:

  - Insert invisible watermarks into structured data files (CSV, XML, and JSON files) and extract the watermarks.
  - Insert visible watermarks into unstructured data files (DOCX, PPTX, XLSX, and PDF files) and open the files on a local host to view the watermarks.
- Dynamic watermarking

  After dynamic watermarking is enabled for DataArts Factory and a dynamic watermarking policy is created in DataArts Security, an invisible dark watermark will be inserted into the sensitive data dumped or downloaded by a user group or role specified in the policy to prevent the sensitive data from being disclosed. The watermark is the first 16 digits from the ID of the IAM user who is attempting to obtain the sensitive data. For details about how to view the IAM user ID, see "Obtaining a Project ID and Account ID" in **(Optional) Obtaining Authentication Information**.

9.5.2 Static Masking Tasks

9.5.2.1 Managing Masking Algorithms

Masking algorithms are mandatory for creating masking policies. The system provides more than 20 built-in masking algorithms. If you want to use these algorithms, you need to configure their parameters. If the built-in algorithms cannot meet your needs, you can create algorithms.

This section describes built-in masking algorithms and how to create masking algorithms.

Built-in Masking Algorithms

The following table lists the masking algorithms.

Table 9-32 Algorithm types

Type	Description	Scenario	Example	Original	Masked
Hash	Convert data by using hash functions such as password salting and keys.	Used to anonymize structured and unstructured data.	HMAC-SHA256 hash	460031234567890	A34329AE133C48C

Type	Description	Scenario	Example	Original	Masked
Cut	Discard the last few numbers of an attribute to ensure data fuzziness.	Used to anonymize structured and unstructured data. For example, it can be used to anonymize identifiers and quasi-identifiers.	Cut the last four numbers.	18012345678	1801234
Mask	Replace some characters in an attribute with special characters. Example: *	Used to anonymize structured and unstructured data, such as identifiers and quasi-identifiers.	Mask the last four numbers.	18012345678	1801234****
Encryption	Invoke the built-in encryption algorithms of GaussDB(DWS) and Hive to encrypt data.	There are strict restrictions on the data to be encrypted.	AES	98	2bd806c97f0e00af1a1fc3328fa763a9269723c8db8fac4f93af71db186d6e

DataArts Security provides the following built-in masking algorithms. Before selecting an algorithm, you can use the algorithm configuration and testing functions to check whether the algorithm suits your needs.

Table 9-33 Built-in algorithms

Type	Name	Description	Configurable
Hash	HMAC-SHA256 hash	Use the HMAC-SHA256 algorithm for hash processing.	A salt value and a key can be configured. **NOTE** • Before using the algorithm, you must configure a key. • You need to set a salt value rather than use the secure random number provided by the system. Pay attention to the risks.
	SHA-256	Use the SHA-256 algorithm for hash processing.	A salt value can be configured. **NOTE** You need to set a salt value rather than use the secure random number provided by the system. Pay attention to the risks.
Cut	Value truncation	Retain x digits before the decimal point and replace the x-1 digits from the first digit before the decimal point and the digits after the decimal point with 0. For example, if x is 3, 1234 is truncated to 1200, 999.999 is truncated to 900, and 10.7 is truncated to 0.	The number of digits before the decimal point can be configured.
	Date truncation	Truncate a specified date.	The date format and masking range can be configured.
Mask	Masking of specified GaussDB(DWS) columns	Masks specified columns in GaussDB(DWS). This algorithm can be used only when both the source and destination of a static masking task are GaussDB(DWS) and the execution engine is GaussDB(DWS).	Not supported

Type	Name	Description	Configurable
	Masking with specified characters for GaussDB(DWS)	Replaces the characters from the start to end position with specified characters. This algorithm can be used only when both the source and destination of a static masking task are GaussDB(DWS) and the execution engine is GaussDB(DWS).	The start position, end position, and mask flag can be configured.
	Masking with specified digits for GaussDB(DWS)	Replaces the characters from the start to end position with specified digits. This algorithm can be used only when both the source and destination of a static masking task are GaussDB(DWS) and the execution engine is GaussDB(DWS).	The start position, end position, and mask flag can be configured.
	ID masking	Masks an ID card No.	Not supported
	Bank card No. masking	Masks a bank card No.	Not supported
	Email masking	Masks email information.	Not supported
	Mobile equipment identity masking	Masks the device code, such as IMEI, MEDI, and ESN.	The type can be configured.
	IPv6 masking	Masks an IPv6 address.	Not supported
	IPv4 masking	Masks an IPv4 address.	Not supported
	MAC address masking	Masks a MAC address.	Not supported
	Phone No. masking	Masks a phone number.	Not supported

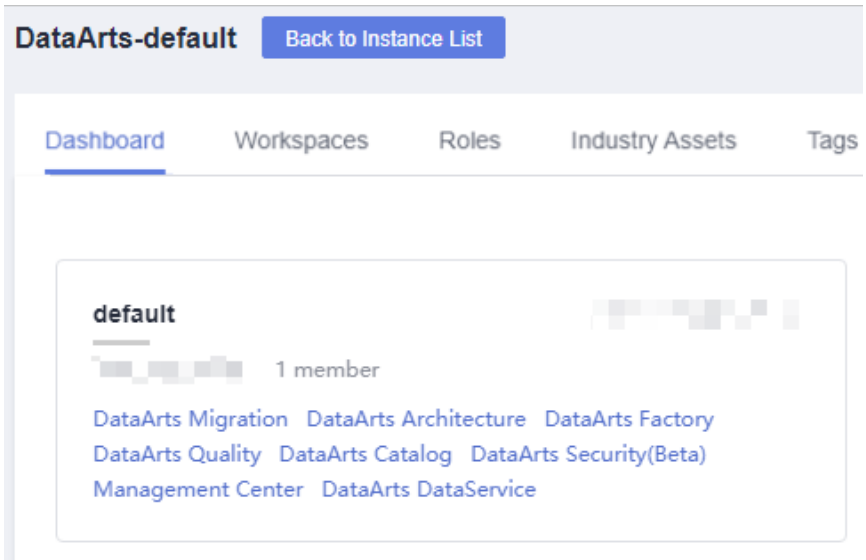
Type	Name	Description	Configurable
	Date type masking	Masks a specified date format, such as ISO, EUR, and USA.	The date format and masking range can be configured.
	Masking *X* to *Y*	Masks the characters from *X* to *Y* of a string.	*X* and *Y* can be configured.
	Retaining *X* to *Y*	Retains the characters from *X* to *Y* of a string.	*X* and *Y* can be configured.
	Masking first *n* and last *m* characters	Masks the first *n* and last *m* characters of a string.	*n* and *m* can be configured.
	Retaining first *n* and last *m* characters	Retains the first *n* and last *m* characters of a string.	*n* and *m* can be configured.
Encryption	GaussDB(DWS) column encryption	The symmetric cryptographic algorithm gs_encrypt_aes128(encryptstr,keystr) provided by GaussDB (DWS) is invoked to encrypt DWS data columns. This algorithm uses keystr as the key to encrypt the encryptstr character string and returns the encrypted character string. Note the following: ● This algorithm takes effect only when the destination of the masking task is GaussDB (DWS). ● When SQL decryption is executed after encryption, the decryption result can be correctly returned only when all data is successfully decrypted. Otherwise, the decryption fails.	The key can be configured. The key length ranges from 1 byte to 16 bytes. **NOTE** Before using the algorithm, you must configure a key.

Type	Name	Description	Configurable
	Hive column encryption	Invokes the Hive column encryption function provided by MRS to encrypt and decrypt Hive data columns. Cryptographic algorithms AES and SMS4 are supported. Note the following: • This algorithm takes effect only when the target of the masking task is Hive. • Column encryption can be performed in HDFS tables of only the TextFile and SequenceFile file formats. • The Hive column encryption does not support views and the Hive over HBase scenario.	The encryption type can be configured.

Creating a Masking Algorithm

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

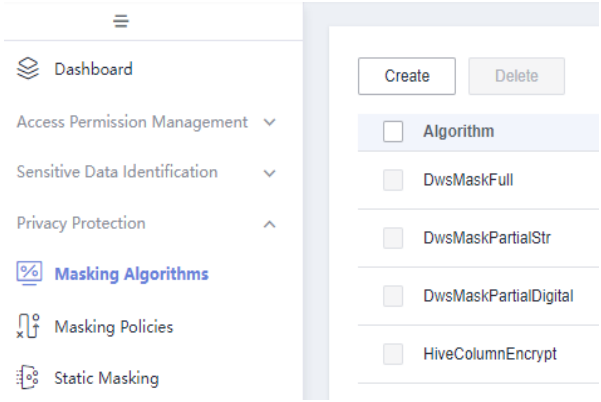
Figure 9-177 DataArts Security



Step 2 In the left navigation pane, choose **Masking Algorithms**.

Step 3 Click **Create**.

Figure 9-178 Creating a masking algorithm



Step 4 Set the parameters listed in **Table 9-34** and click **OK**.

Figure 9-179 Configuring algorithm parameters

The following table lists the masking algorithm parameters.

Table 9-34 Parameters for the masking algorithm

Parameter	Description
*Algorithm	Name of the algorithm to be created. It can contain a maximum of 64 characters and can consist of only letters, digits, and underscores.
Description	Brief description of the algorithm. It can contain a maximum of 255 characters.
*Algorithm Template	Built-in algorithm template used to customize the algorithm. For details about the available algorithm types and algorithms, see **Built-in Masking Algorithms**.

----End

Related Operations

- Editing an algorithm: On the **Masking Algorithms** page, locate an algorithm and click **Edit** in the **Operation** column.

  The parameters that can be edited vary depending on the algorithm type.
- Testing an algorithm: On the **Masking Algorithms** page, locate an algorithm and click **Test** in the **Operation** column.

  NOTE

  Before using an algorithm, you are advised to test it to ensure that it meets your needs.

  Whether the test function is available varies depending on the algorithm type.
- Deleting algorithms: On the **Masking Algorithms** page, locate an algorithm and click **Delete** in the **Operation** column. To delete multiple algorithms, select them and click **Delete** above the list.

  Built-in algorithms cannot be deleted. Custom algorithms that are used by masking policies or specified column masking cannot be deleted. To delete such algorithms, cancel the reference first.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.

9.5.2.2 Managing Masking Policies

In business activities, some enterprise departments need to analyze data for operations. In this case, data must be accessible to these departments even if it is sensitive. To meet this requirement and prevent data leakage, you can create data masking policies to mask sensitive data.

This section describes how to manage the masking policies for static masking tasks.

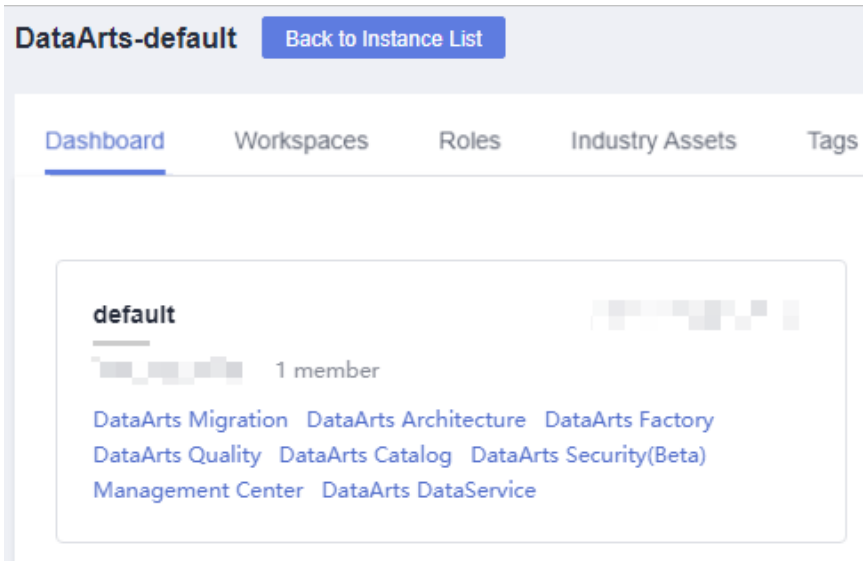
Prerequisites

- A sensitive data identification rule has been created. For details, see **Creating Identification Rules**.
- A built-in or custom masking algorithm has been created. For details, see **Managing Masking Algorithms**.

Creating a Data Masking Policy

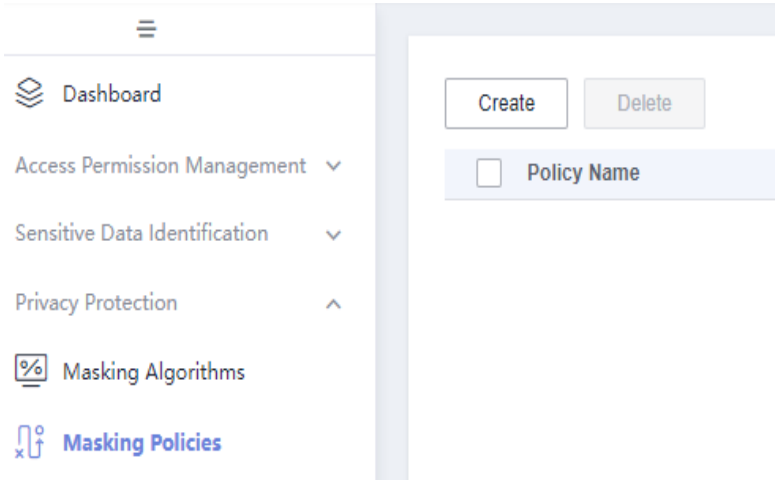
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-180 DataArts Security



Step 2 Choose **Masking Policies** from the left navigation bar, and click **Create** in the upper part of the displayed page.

Figure 9-181 Creating a data masking policy



Step 3 In the displayed dialog box, set the parameters listed in **Table 9-35** and click **OK**.

Figure 9-182 Creating a data masking policy

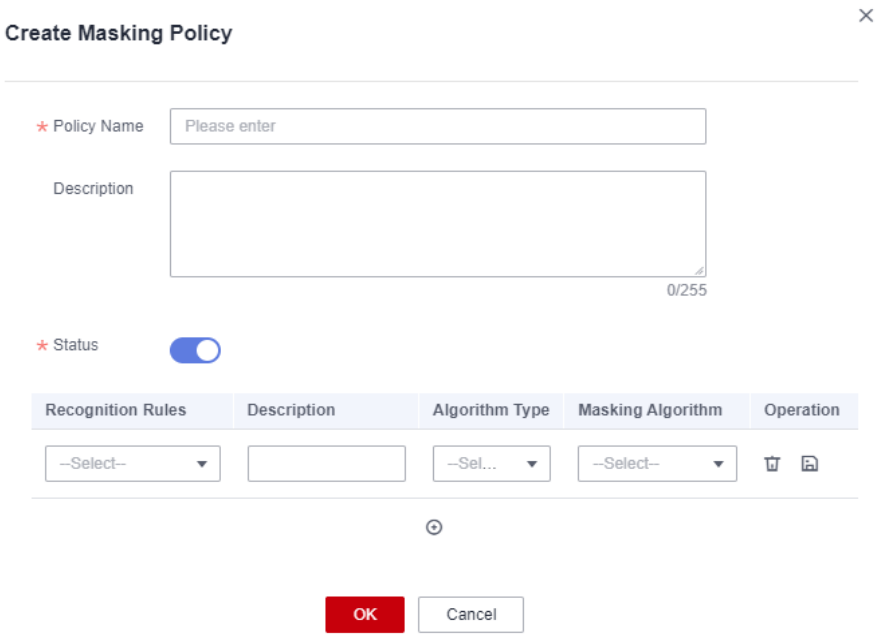


Table 9-35 Parameters

Parameter	Description
*Policy Name	The name of the policy to be created. Policy names can include only letters, numbers, and underscores (_) and cannot exceed 64 characters.
Description	A description of the policy to be created, which can contain a maximum of 255 characters.
*Status	If the status switch is turned on, the policy is available. If the status switch is turned off, the policy cannot be used.
*Recognition Rules and Masking Algorithm	Sensitive data identification rule and the corresponding masking algorithm • ***Recognition Rules**: Select a data identification rule. For details, see **Creating Identification Rules**. • **Description**: Enter a description of the rule. • ***Algorithm Type**: Select an algorithm type. For details, see **Table 9-33**. • ***Masking Algorithm**: Select an algorithm of the selected type. For details, see **Table 9-33**. **NOTE** Before using the following masking algorithms, you must configure keys: • HMAC-SHA256 hash algorithm • DWS column encryption algorithm For more restrictions on different masking algorithms, see **Managing Masking Algorithms**.

----End

Related Operations

- Editing a masking policy: On the **Masking Policies** page, locate a policy and click **Edit** in the **Operation** column.
- Setting the masking policy status: A masking policy is enabled by default. If a data masking policy is disabled, it cannot be used by static data masking tasks.

  To change the status of a data masking policy, click  or  to enable or disable the policy.

  NOTE

  Masking policies used by static masking tasks cannot be disabled.
- Deleting masking policies: On the **Masking Policies** page, locate a policy and click **Delete** in the **Operation** column. To delete multiple policies, select them and click **Delete** above the list.

  Policies used by static masking tasks cannot be deleted. To delete such policies, modify the reference relationship first.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.

9.5.2.3 Managing Static Masking Tasks

This section describes how to create a static masking task. For the source and destination types that support static masking, see **Reference: Static Data Masking Scenarios**.

Static data masking prevents private data leakage, and ensures regulatory compliance as well as data security for enterprises. Sensitive data is masked, truncated, and hashed based on the abundant and effective built-in masking algorithms, and the processed data can be written to the target data table. For security purpose, it is the target data table that can be used to provide services for external requirements.

Prerequisites

- Static masking tasks rely on masking policies. The prerequisites are as follows:
  - A built-in or custom masking algorithm has been created. For details, see **Managing Masking Algorithms**.
  - A masking policy has been created. For details, see **Creating a Data Masking Policy**.
  - A sensitive data discovery task has been created for the data tables to be masked. For details, see **Creating a Sensitive Data Discovery Task**.
  - The sensitive data status has been changed to valid on the **Sensitive Data Distribution** page. For details, see **Viewing Sensitive Data Distribution**.

- For static masking tasks using the DLI engine, the following OBS permissions have been granted to the **dlg_agency**. For details, see **Authorizing dlg_agency**.

```
obs:bucket:HeadBucket
obs:bucket:CreateBucket
obs:object:PutObject
obs:object:DeleteObject
obs:bucket:ListBucket
obs:object:GetObject
obs:bucket:GetEncryptionConfiguration
obs:bucket:PutEncryptionConfiguration
```

Constraints

- For a static masking task using the DLI engine, the running parameters need to be stored in an OBS bucket. After the task is complete or fails, the task running parameter file is deleted.
  - For a same-source static masking task using the DLI engine, the running parameters are stored in the workspace log bucket named **dlf-log-*{Project id}*** by default.
  - For a cross-source static masking task using the DLI engine, the running parameters are stored in the encrypted user bucket named **dls-dli-*{projectId}*** that is automatically created.

  Therefore, before performing static masking using the DLI engine, you must grant the following OBS permissions to the **dlg_agency**. For details, see **Authorizing dlg_agency**.

```
obs:bucket:HeadBucket
obs:bucket:CreateBucket
obs:object:PutObject
obs:object:DeleteObject
obs:bucket:ListBucket
obs:object:GetObject
obs:bucket:GetEncryptionConfiguration
obs:bucket:PutEncryptionConfiguration
```

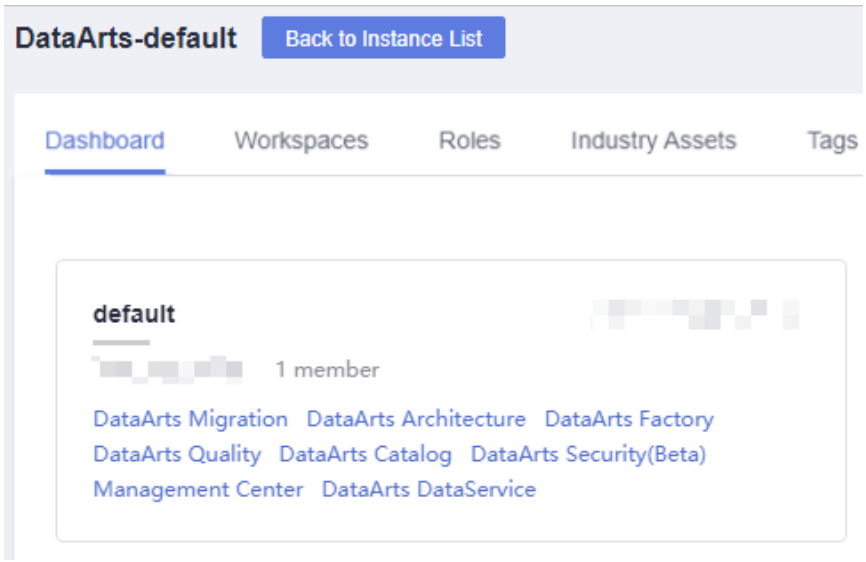
- For a static masking task using the DLI engine, if the source or destination is GaussDB(DWS), enable network communications between the DLI Spark common queue and GaussDB(DWS). Otherwise, the static masking task will fail. For details, see **Configuring the Connection Between a DLI Queue and a Data Source in a Private Network** or **Configuring the Connection Between a DLI Queue and a Data Source in the Internet**.
- Kerberos authentication must be enabled for the MRS cluster where MRS Hive is located, and the Spark component must be installed for the MRS cluster.
- For a static masking task using the MRS engine, if the source or destination is GaussDB(DWS), configure an agency for the MRS cluster by referring to **Reference: Authorizing and Binding an Agency** and ensure that the outbound rule of the MRS cluster's security group meets the following requirements. Otherwise, the static masking task will fail.
  - Protocol: TCP
  - Port: 80
  - Destination: 169.254.0.0/16
- For a static masking task using the MRS engine, if either the source or destination is GaussDB(DWS), the following data types are supported. If there is data of other unsupported types, the static masking task will fail.

  - tinyint
  - smallint
  - int
  - bigint
  - decimal
  - double
  - float
  - boolean
  - string
  - timestamp
- A same-source static masking task using the GaussDB(DWS) engine does not support cross-database masking. That is, the source and destination data tables must be in the same database.
- If the source or destination of a static masking task is DLI, data tables in the DLI default database cannot be masked.
- If **Dataset Scope** is set to **Incremental** for a static masking task, **Timestamp** or **Date** needs to be selected for **Time Field**.

Create a Static Masking Task

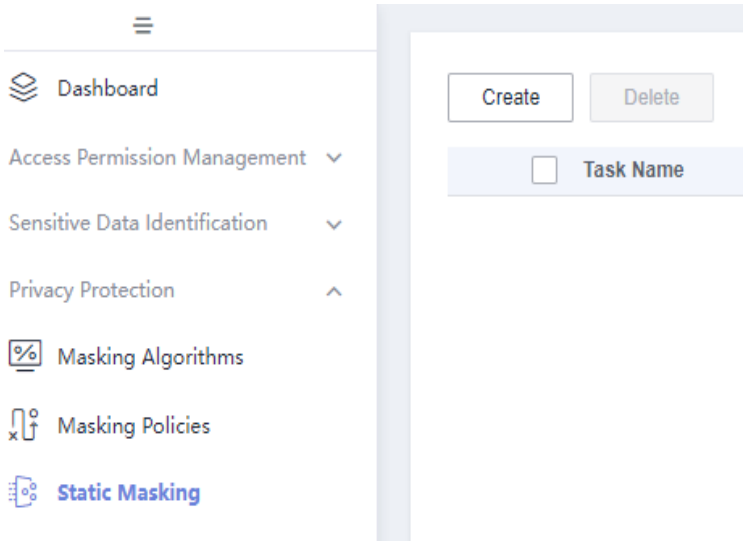
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-183 DataArts Security



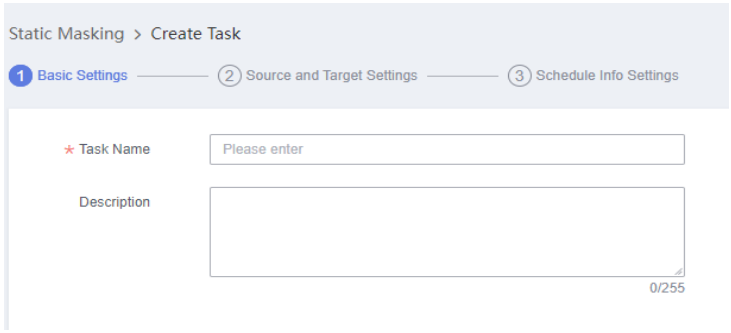
Step 2 In the left navigation pane, choose **Static Masking**. In the right pane, click **Create**.

Figure 9-184 Creating a static masking task



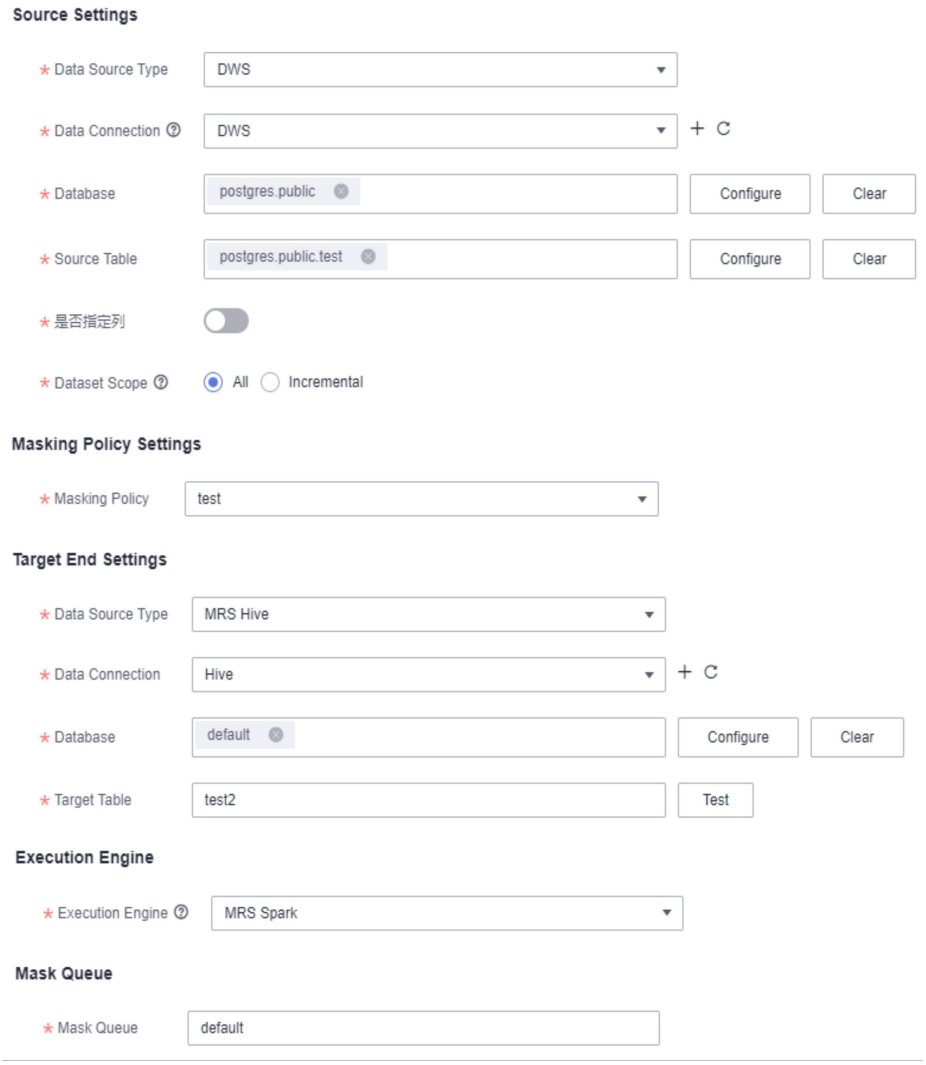
Step 3 In the displayed dialog box, set **Task Name** and **Description** and click **Next**.

Figure 9-185 Configuring basic information



Step 4 Configure the source and destination parameters. For parameter details, see **Table 9-36**.

Figure 9-186 Configuring the masking task



The following table lists the parameters of the masking task.

Table 9-36 Parameters of the masking task

Parameter	Description
Source Settings	
*Data Source Type	**DLI**, **DWS** and **MRS Hive** are supported.
*Data Connection	Select a data connection that has been created in Management Center. If no data connection is available, create one by referring to **Creating a Data Connection**.
*SQL Queue	This parameter is mandatory if **Data Source Type** is set to **DLI**.

Parameter	Description
*Database	Click **Configure** to select the database whose data is to be masked. Data tables in the DLI default database cannot be masked.
*Source Table	Click **Configure** to select the table whose data is to be masked.
*Specify Column	Whether to specify the columns to mask. If this function is enabled, you can configure masking algorithms for specified columns in the source table. You can configure different masking algorithms for multiple columns. **NOTE** Once saved, this option cannot be changed.
*Column	This parameter is mandatory when **Specify Column** is enabled. If you want to mask a column, you must select the column and select a masking algorithm. If you only select the masking algorithm, no column will be masked. **NOTE** Before using the following masking algorithms, you must configure keys: • HMAC-SHA256 hash algorithm • DWS column encryption algorithm For more restrictions on different masking algorithms, see **Managing Masking Algorithms**.
*Dataset Scope	If **Dataset Scope** is set to **Incremental**, you can set **Time Field** to **Timestamp** or **Date**. Generally, the masking task is scheduled once if this parameter is set to **All** and is scheduled periodically if this parameter is set to **Incremental**.
*Time Field	If **Dataset Scope** is set to **Incremental**, you can set this parameter to **Timestamp** or **Date**.
Masking Policy Settings	
*Masking Policy	This parameter is configurable only when no column is specified. Select a created masking policy from the drop-down list.
Target End Settings	
*Data Source Type	Select the storage type for the masked data. **Table 9-38** lists the supported masking scenarios.
*Data Connection	Select a data connection that has been created in Management Center. If no data connection is available, create one by referring to **Creating a Data Connection**.
*SQL Queue	This parameter is mandatory if **Data Source Type** is set to **DLI**.

Parameter	Description
*Database	Click **Configure** to select the database for storing the masked data. Data tables in the DLI default database cannot be masked.
*Target Table	Enter a unique table name. The table is automatically created when the table name entered does not exist. Click **Test** to check whether the target table can be used. Otherwise, you cannot proceed to the next step.
Execution Engine	
*Execution Engine	Select the engine that runs the masking task. **Table 9-38** lists the supported engines and precautions in different masking scenarios.
Masking Queue	
* Mask Queue	Select a queue in the DLI or MRS engine. • If the execution engine is DLI, select a DLI Spark common queue. For a static masking task using the DLI engine, if the source or destination is GaussDB(DWS), enable network communications between the DLI Spark common queue and GaussDB(DWS). Otherwise, the static masking task will fail. For details, see **Configuring the Connection Between a DLI Queue and a Data Source in a Private Network** or **Configuring the Connection Between a DLI Queue and a Data Source in the Internet**. • If the execution engine is MRS, you need to enter the MRS tenant queue. To view available queues, you can click a cluster name in the cluster list on the MRS console to go to the cluster details page and click the **Tenants** tab and then the **Queue Configuration** tab.

Step 5 Click **Next** and configure scheduling.

- If **Dataset Scope** is set to **All**, **Repeat** can be only set to **Once**.
- If **Dataset Scope** is set to **Incremental**, **Repeat** can be set to **Once** or **On Schedule**.

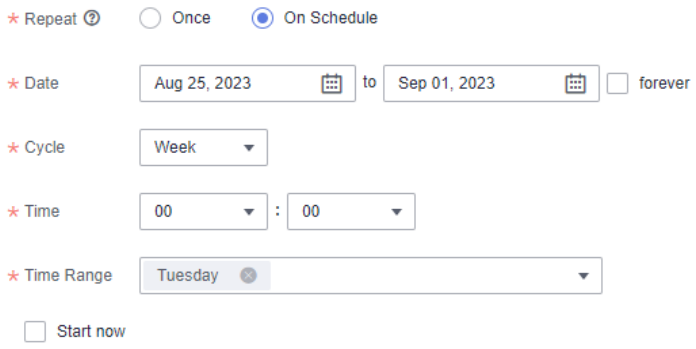
If you set **Repeat** to **On Schedule**, set the parameters listed in **Table 9-37**.

Table 9-37 Parameters for periodic scheduling

Parameter	Description
*Date	Period during which the task takes effect.

Parameter	Description
*Cycle	The frequency at which a task is executed. The options are: • **minutes**: Select the scheduling start time and end time, and set the interval in minutes. • **hours**: Select the scheduling start time and end time, and set the interval in hours. • **Day**: Set the scheduling time everyday. • **Week**: Select a day in a week and set the specific time to start scheduling. • **Month**: Select a day in a month and set the specific time to start scheduling. For example, you can set **Cycle** to **Week**, **Time** to **15:52**, and **Time Range** to **Tuesday**. In this case, the task is executed at 15:52 every Tuesday within the configured date range.
Start now	If you select **Start now**, the task is scheduled immediately.

Figure 9-187 Setting parameters for periodic scheduling



Step 6 After all settings are complete, click **OK**.

----End

Related Operations

- Editing a task: On the **Static Masking** page, locate a task and click **Edit** in the **Operation** column.

  A task in the **Scheduling** state cannot be edited.
- Deleting tasks: On the **Static Masking** page, locate a task, click **More** in the **Operation** column, and select **Delete**. To delete multiple tasks at a time, select the tasks and click **Delete** above the task list.

  A task in the **Scheduling** state cannot be deleted.

NOTE

The deletion operation cannot be undone. Exercise caution when performing this operation.

- Running or scheduling a task: On the **Static Masking** page, locate a task and click **Run** in the **Operation** column or click **More** in the **Operation** column and select **Start**.

  You can determine whether a task is scheduled once or repeatedly based on the scheduling period.

- Viewing running instance logs: On the **Static Masking** page, locate a task and click to expand instances. Then click **View Log**.

  If a task fails to be executed, you can locate the failure cause based on logs, rectify the fault, and try the task again. If the fault persists, contact technical support.

Reference: Authorizing and Binding an Agency

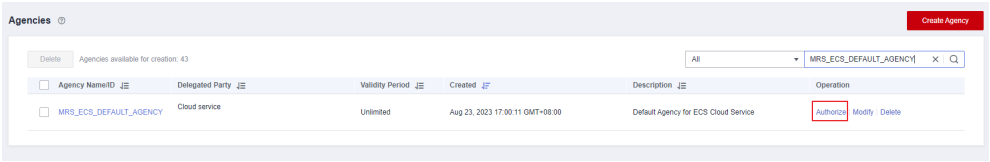
Step 1 Log in to the IAM console.

Step 2 Choose **Agencies**. In the agency list, locate the preset **MRS_ECS_DEFAULT_AGENCY** agency and click **Authorize**.

NOTE

If the preset **MRS_ECS_DEFAULT_AGENCY** agency is not found, you can buy an MRS cluster and select the **MRS_ECS_DEFAULT_AGENCY** agency in advanced settings. When the MRS cluster creation starts, the **MRS_ECS_DEFAULT_AGENCY** agency is automatically generated.

Figure 9-188 Authorizing an agency

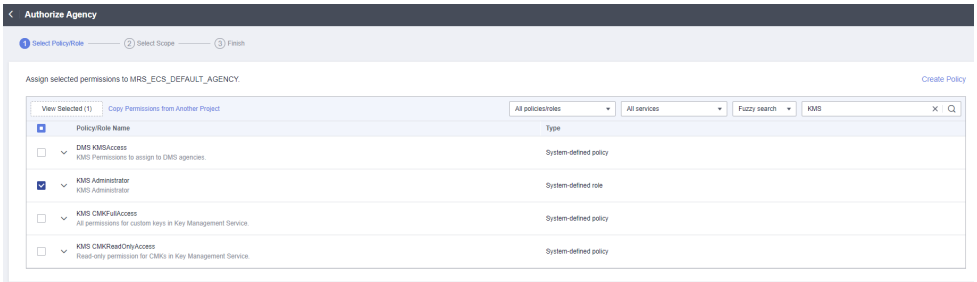


Step 3 In the search box, enter **KMS** and select the **KMS Administrator** policy.

NOTE

The minimum permission required by the **MRS_ECS_DEFAULT_AGENCY** is **kms:cmk:decrypt**. In addition to directly granting the **KMS Administrator** policy, you can create a custom policy with the **kms:cmk:decrypt** permission of the KMS on the IAM console and grant the policy to the **MRS_ECS_DEFAULT_AGENCY**.

Figure 9-189 Selecting permissions

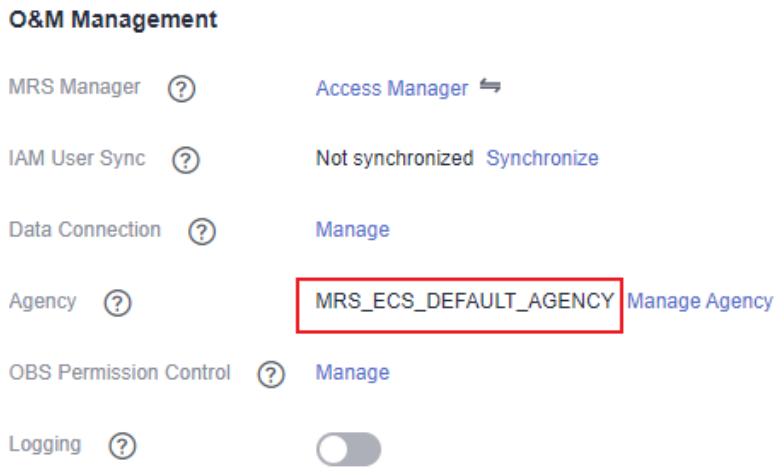


Step 4 After selecting the permission, click **Next** to set the authorization scope. In this example, retain the default settings and click **OK** to complete the authorization.

Step 5 On the MRS management console, choose **Clusters** > **Active Clusters**. Click the name of the target cluster to go to the cluster details page.

Step 6 On the **Dashboard** page, locate the **O&M Management** area and check that the cluster has been bound to the **MRS_ECS_DEFAULT_AGENCY** agency. If the cluster is not bound to the **MRS_ECS_DEFAULT_AGENCY** agency, you need to manually select the **MRS_ECS_DEFAULT_AGENCY** agency.

Figure 9-190 Binding an agency



----End

Reference: Static Data Masking Scenarios

Table 9-38 lists the static masking scenarios supported by privacy protection.

Table 9-38 Static masking scenarios

Data Source (Source)	Data Source (Target)	Computing Engine	Description
Data Lake Insight (DLI)	Data Lake Insight (DLI)	DLI Spark common queue	None

Data Source (Source)	Data Source (Target)	Computing Engine	Description
	GaussDB(DWS)	DLI Spark common queue	• For a static masking task using the DLI engine, if the source or destination is GaussDB(DWS), enable network communications between the DLI Spark common queue and GaussDB(DWS). Otherwise, the static masking task will fail. For details, see **Configuring the Connection Between a DLI Queue and a Data Source in a Private Network** or **Configuring the Connection Between a DLI Queue and a Data Source in the Internet**.

Data Source (Source)	Data Source (Target)	Computing Engine	Description
GaussDB(DWS)	DWS	• GaussDB(DWS) cluster • MRS cluster • DLI Spark common queue	**GaussDB(DWS) engine:** • A same-source static masking task using the GaussDB(DWS) engine does not support cross-database masking. That is, the source and destination data tables must be in the same database. **MRS engine:** • Kerberos authentication must be enabled for the MRS cluster where MRS Hive is located, and the Spark component must be installed for the MRS cluster. • For a static masking task using the MRS engine, if the source or destination is GaussDB(DWS), configure an agency for the MRS cluster by referring to **Reference: Authorizing and Binding an Agency** and ensure that the outbound rule of the MRS cluster's security group meets the following requirements. Otherwise, the static masking task will fail. – Protocol: TCP – Port: 80 – Destination: 169.254.0.0/16 **DLI engine:** • For a static masking task using the DLI engine, if the source or destination is GaussDB(DWS), enable network communications between the DLI Spark common queue and GaussDB(DWS). Otherwise, the static masking task will fail. For details, see **Configuring the Connection Between a DLI Queue and a Data Source in a Private Network** or **Configuring the Connection Between a DLI Queue and a Data Source in the Internet**.

Data Source (Source)	Data Source (Target)	Computing Engine	Description
	MRS Hive	MRS cluster where MRS Hive is located	• Kerberos authentication must be enabled for the MRS cluster where MRS Hive is located, and the Spark component must be installed for the MRS cluster. • For a static masking task using the MRS engine, if the source or destination is GaussDB(DWS), configure an agency for the MRS cluster by referring to **Reference: Authorizing and Binding an Agency** and ensure that the outbound rule of the MRS cluster's security group meets the following requirements. Otherwise, the static masking task will fail. – Protocol: TCP – Port: 80 – Destination: 169.254.0.0/16 • For a static masking task using the MRS engine, if either the source or destination is GaussDB(DWS), the following data types are supported. If there is data of other unsupported types, the static masking task will fail. – tinyint – smallint – int – bigint – decimal – double – float – boolean – string – timestamp

Data Source (Source)	Data Source (Target)	Computing Engine	Description
	Data Lake Insight (DLI)	DLI Spark common queue	• For a static masking task using the DLI engine, if the source or destination is GaussDB(DWS), enable network communications between the DLI Spark common queue and GaussDB(DWS). Otherwise, the static masking task will fail. For details, see **Configuring the Connection Between a DLI Queue and a Data Source in a Private Network** or **Configuring the Connection Between a DLI Queue and a Data Source in the Internet**.
MRS Hive	MRS Hive	MRS cluster where the source MRS Hive is located	• Kerberos authentication must be enabled for the MRS cluster where MRS Hive is located, and the Spark component must be installed for the MRS cluster.

Data Source (Source)	Data Source (Target)	Computing Engine	Description
	GaussDB(DWS)	MRS cluster where MRS Hive is located	• Kerberos authentication must be enabled for the MRS cluster where MRS Hive is located, and the Spark component must be installed for the MRS cluster. • For a static masking task using the MRS engine, if the source or destination is GaussDB(DWS), configure an agency for the MRS cluster by referring to **Reference: Authorizing and Binding an Agency** and ensure that the outbound rule of the MRS cluster's security group meets the following requirements. Otherwise, the static masking task will fail. – Protocol: TCP – Port: 80 – Destination: 169.254.0.0/16 • For a static masking task using the MRS engine, if either the source or destination is GaussDB(DWS), the following data types are supported. If there is data of other unsupported types, the static masking task will fail. – tinyint – smallint – int – bigint – decimal – double – float – boolean – string – timestamp

9.5.3 Dynamic Masking Tasks

9.5.3.1 Managing Dynamic Masking Policies

After a dynamic masking policy is created in DataArts Security, the system synchronizes the policy to the data source. The data source dynamically masks

data columns based on specified rules. When the users and user groups specified in the policy access sensitive data, the system returns the data that is dynamically masked by the data source to protect sensitive data from being disclosed.

Note that dynamic masking policies configured for a DataArts Studio instance are visible to and take effect for all the workspaces of the instance.

Prerequisites

- Before creating MRS Hive data masking policies, you have created an MRS Ranger data connection. For details, see **Creating a Data Connection**.
- Before creating GaussDB(DWS) data masking policies, you have created a GaussDB(DWS) connection. For details, see **Creating a Data Connection**. The account in the GaussDB(DWS) connection must have the GRANT permission of the target table. (By default, only the owner of a database object or system administrator can run the **GRANT** command to grant the object permissions to other users.)
- Dynamic masking policies for MRS Hive and GaussDB(DWS) need to be associated with data sources for specified users or user groups. Therefore, you need to synchronize user information from IAM to data sources first. For details, see **Synchronizing IAM Users to the Data Source**.
- If you want to use the current user identity authentication to make the dynamic masking policy take effect during script execution and job tests in DataArts Factory, you need to enable permission applications by following the instructions in **Enabling Fine-grained Authentication**.
- If you want to view sensitive fields during the creation of a data masking policy, you need to create a sensitive data discovery task in advance and change the statuses of sensitive data fields to valid on the **Sensitive Data Distribution** page. For details, see **Discovering Sensitive Data** and **Viewing Sensitive Data Distribution**.

Constraints

- Only the DAYU Administrator, Tenant Administrator, or data security administrator can create, modify, or delete dynamic masking policies. Other common users do not have permission to perform these operations.
- Currently, dynamic masking policies support only MRS Hive and GaussDB(DWS) data sources, and do not support GaussDB(DWS) logical clusters. In addition, accounts in GaussDB(DWS) data connections must have the GRANT permission of the table to be masked. (After a database object is created, only the object owner or system administrator can grant the object permissions to other users using the **GRANT** command by default.)
- A table can be associated with only one dynamic data masking policy. Policies take effect only after they are synchronized successfully.
- During dynamic masking of MRS Hive data, MRS Ranger allows you to configure different rules for the same column, and the rules are matched in the sequence of their configuration time. Therefore, you can configure multiple masking policies for different content in the same cluster, database, table, and column.
- Dynamic masking policies for MRS Hive and GaussDB(DWS) need to be associated with data sources for specified users or user groups. Therefore, you

need to synchronize user information from IAM to data sources first. For details, see **Synchronizing IAM Users to the Data Source**.

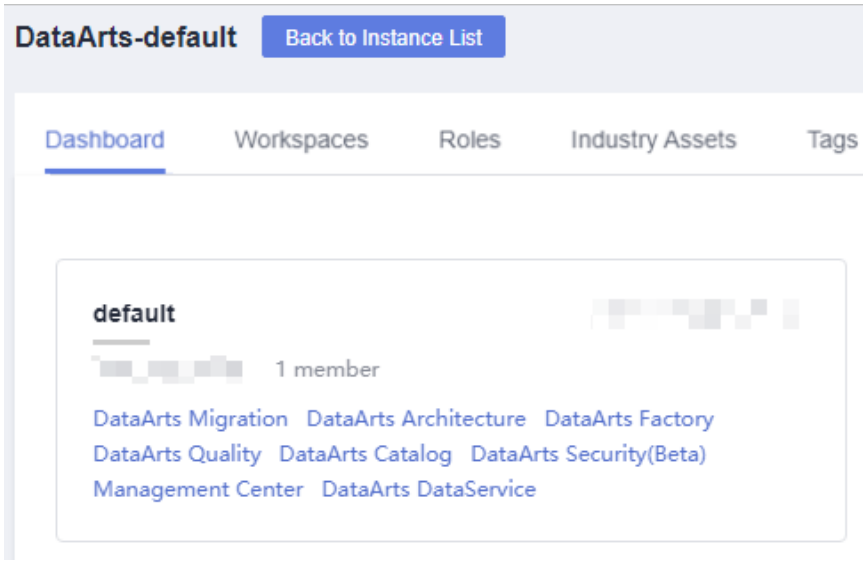
- **Table 9-40** lists the masking rules supported by the MRS service. For Chinese characters, only null and hash masking are supported. If other masking methods are selected, masking does not take effect.

  The SM3, Custom/Show First x and Last y Characters, and Custom/Mask First x and Last y Characters masking rules for the MRS Hive data source are not provided by the MRS Ranger component. Instead, they are implemented by UDF-defined functions. Therefore, to use any of the three masking rules, you must upload the JAR package on which the algorithm depends to the MRS cluster, and grant the UDF creation permission to the account in the Ranger data connection and the UDF usage permission to all users in advance. For details, see **Reference: Configuring UDF-related Permissions in the Ranger Component**.
- **Table 9-41** lists the masking rules supported by GaussDB(DWS). Chinese characters cannot be masked. If you mask data that contains Chinese characters, garbled characters may be displayed.

Creating a Dynamic Masking Policy

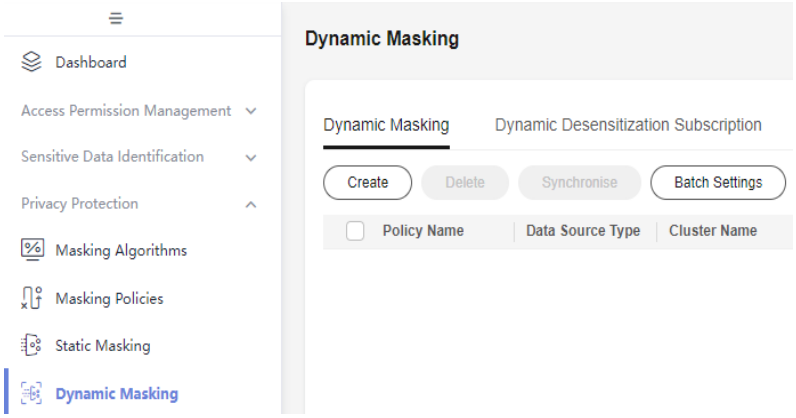
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-191 DataArts Security



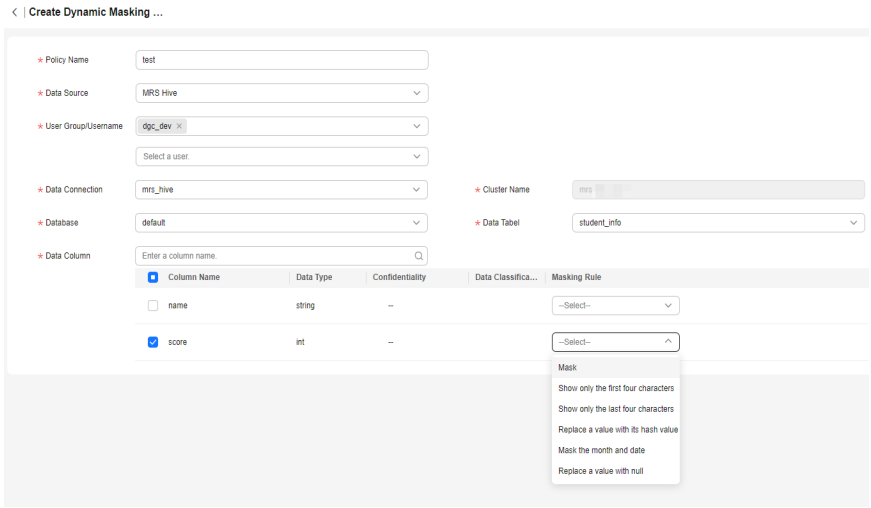
Step 2 In the left navigation pane, choose **Dynamic Masking**.

Figure 9-192 Dynamic Masking



Step 3 Click **Create** and set the parameters listed in **Table 9-39**.

Figure 9-193 Setting parameters for the dynamic masking policy



The following table lists the parameters.

Table 9-39 Policy parameters

Parameter	Description
*Policy Name	Unique identifier of the dynamic masking policy. It must be unique in a DataArts Studio instance. To facilitate policy management, you are advised to include the object to be masked and masking rule in the name.
*Data Source Type	Currently, only **MRS Hive** and **DWS** are supported.
MRS Hive	

Parameter	Description
*User Group/ Username	User or user group in the current workspace members. When a specified object queries or exports sensitive data from DataArts Factory, the system dynamically masks the sensitive data to protect the sensitive data from being disclosed.
*Data Connection	If no data connection is available, create one by referring to **Creating a Data Connection**.
*Cluster Name	You do not need to set this parameter. The data source cluster in the data connection is automatically selected.
*Database	Database where the sensitive data is stored
*Data Table	Data table where the sensitive data is stored
*Data Column	Select one or more columns to be masked and select a proper masking rule for each column based on the data type. Supported data masking rules vary depending on the data type of each data source. For details, see **Reference: Dynamic Masking Rules**. If sensitive data discovery has been performed on the selected columns and the statuses of the sensitive data fields are valid, the data security levels and classifications are displayed in the **Data Column** area.
DWS	
*User Group/ Username	User or user group in the current workspace members. When a specified object queries or exports sensitive data from DataArts Factory, the system dynamically masks the sensitive data to protect the sensitive data from being disclosed.
*Data Connection	If no data connection is available, create one by referring to **Creating a Data Connection**.
*Cluster Name	You do not need to set this parameter. The data source cluster in the data connection is automatically selected.
*Database	Database where the sensitive data is stored
*schema	Schema where the sensitive data is stored
*Data Table	Data table where the sensitive data is stored
*Data Column	Select one or more columns to be masked and select a proper masking rule for each column based on the data type. Supported data masking rules vary depending on the data type of each data source. For details, see **Reference: Dynamic Masking Rules**. If sensitive data discovery has been performed on the selected columns and the statuses of the sensitive data fields are valid, the data security levels and classifications are displayed in the **Data Column** area.

Step 4 After setting all required parameters, click **OK**. After the dynamic masking policy is created, you need to click **Synchronize** to synchronize the policy to the data source.

----End

Related Operations

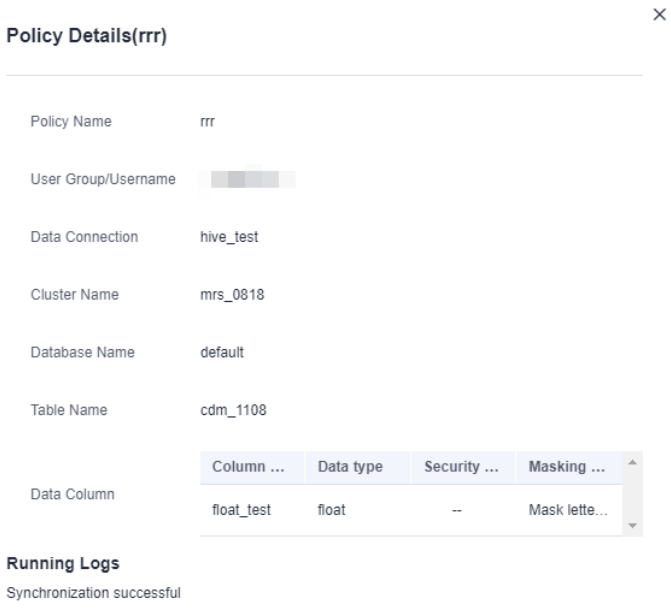
- Synchronizing a policy: On the **Dynamic Masking** page, locate a policy and click **Synchronize** in the **Operation** column to synchronize the policy to the data source. To synchronize multiple policies, select them and click **Synchronize** above the list.

  Policies take effect only after they are synchronized successfully. If the policy synchronization fails, you can view the policy run log in the **policy details** to locate the failure cause. After rectifying the fault, synchronize the policy again. If the synchronization still fails, contact technical support.
- Editing a policy: On the **Dynamic Masking** page, locate a policy and click **Edit** in the **Operation** column.
- Deleting policies: On the **Dynamic Masking** page, locate a policy and click **Delete** in the **Operation** column. In the displayed dialog box, confirm the policy to delete and click **Yes**. To delete multiple policies, select them and click **Delete** above the list.

  NOTE

  Deleted dynamic masking policies are moved to the recycle bin. You can restore them within 30 days. After 30 days, they will be deleted permanently. For details, see **Managing the Recycle Bin**.
- Viewing policy details: On the **Dynamic Masking** page, locate a policy and click its name to view its details. You can also filter policies by **Sync Status**.

Figure 9-194 Viewing policy details



Reference: Dynamic Masking Rules

- MRS Hive dynamic masking rules are provided by MRS Ranger. **Table 9-40** lists the supported rules.

  NOTE

  The SM3, Custom/Show First x and Last y Characters, and Custom/Mask First x and Last y Characters masking rules for the MRS Hive data source are not provided by the MRS Ranger component. Instead, they are implemented by UDF-defined functions. Therefore, to use any of the three masking rules, you must upload the JAR package on which the algorithm depends to the MRS cluster, and grant the UDF creation permission to the account in the Ranger data connection and the UDF usage permission to all users in advance. For details, see **Reference: Configuring UDF-related Permissions in the Ranger Component**.

- GaussDB(DWS) dynamic masking rules are provided by GaussDB(DWS). **Table 9-41** lists the supported rules.

Table 9-40 MRS dynamic masking rules

Data Type	Masking Rule								
	Mask Letters and Digits	Show Only the Last Four Characters	Show Only the First Four Characters	Replace a Value with Its Hash Value	Mask the Month and Date	Replace a Value with Null	SM3	Custom/Show First x and Last y Characters	Custom/Mask First x and Last y Characters
TINYINT	The number of characters remains unchanged. All values are replaced with 1.	No change. The maximum value is 127.	No change. The minimum value is –128.	The value changes to null.	The number of characters remains unchanged. All values are replaced with 1.	The value changes to null.	Not supported	Not supported	Not supported

Data Type	Masking Rule								
	Mask Letters and Digits	Show Only the Last Four Characters	Show Only the First Four Characters	Replace a Value with Its Hash Value	Mask the Month and Date	Replace a Value with Null	SM3	Custom/Show First x and Last y Characters	Custom/Mask First x and Last y Characters
SMALLINT	The number of characters remains unchanged. All values are replaced with 1.	No change. The maximum value is 12767.	No change. The maximum value is –32768.	The value changes to null.	The number of characters remains unchanged. All values are replaced with 1.	The value changes to null.	Not supported	Not supported	Not supported
INT	The number of characters remains unchanged. All values are replaced with 1.	The last four characters are shown.	The first four characters are shown.	The value changes to null.	The number of characters remains unchanged. All values are replaced with 1.	The value changes to null.	Not supported	Not supported	Not supported

Data Type	Masking Rule								
	Mask Letters and Digits	Show Only the Last Four Characters	Show Only the First Four Characters	Replace a Value with Its Hash Value	Mask the Month and Date	Replace a Value with Null	SM3	Custom/Show First x and Last y Characters	Custom/Mask First x and Last y Characters
BIGINT	The number of characters remains unchanged. All values are replaced with 1.	The last four characters are shown.	The first four characters are shown.	The value changes to null.	The number of characters remains unchanged. All values are replaced with 1.	The value changes to null.	Not supported	Not supported	Not supported
BOOLEAN	The value changes to null.	The value changes to null.	The value changes to null.	The value changes to null.	The value changes to null.	The value changes to null.	Not supported	Not supported	Not supported
FLOAT	The value changes to null.	The value changes to null.	The value changes to null.	The value changes to null.	The value changes to null.	The value changes to null.	Not supported	Not supported	Not supported

Data Type	Masking Rule								
	Mask Letters and Digits	Show Only the Last Four Characters	Show Only the First Four Characters	Replace a Value with Its Hash Value	Mask the Month and Date	Replace a Value with Null	SM3	Custom/Show First x and Last y Characters	Custom/Mask First x and Last y Characters
DOUBLE	The value changes to null.	The value changes to null.	The value changes to null.	The value changes to null.	The value changes to null.	The value changes to null.	Not supported	Not supported	Not supported
STRING	Letters change to x, and digits change to n.	Chinese characters remain unchanged, and letters change to X.	Letters change to X.	The value changes to its hash value of 64 bytes.	Chinese characters remain unchanged with each character occupying one digit, and letters change to X.	The value changes to null.	Encryption using the SM3 algorithm	Shows the first x characters and the last y characters.	Hides the first x characters and the last y characters.
TIMESTAMP	The value changes to null.	The value changes to null.	The value changes to null.	The value changes to null.	The value changes to null.	The value changes to null.	Not supported	Not supported	Not supported

Data Type	Masking Rule								
	Mask Letters and Digits	Show Only the Last Four Characters	Show Only the First Four Characters	Replace a Value with Its Hash Value	Mask the Month and Date	Replace a Value with Null	SM3	Custom/Show First x and Last y Characters	Custom/Mask First x and Last y Characters
CHAR	Letters change to x, and digits change to n.	Letters and digits change to X, and the last four characters are retained (a fixed length with spaces).	Letters and digits change to X, and the first four characters are retained (a fixed length with spaces).	The value changes to its hash value of 64 bytes.	Chinese characters remain unchanged with each character occupying one digit, and letters change to X.	The value changes to null.	Encryption using the SM3 algorithm	Shows the first x characters and the last y characters.	Hides the first x characters and the last y characters.

Data Type	Masking Rule								
	Mask Letters and Digits	Show Only the Last Four Characters	Show Only the First Four Characters	Replace a Value with Its Hash Value	Mask the Month and Date	Replace a Value with Null	SM3	Custom/Show First x and Last y Characters	Custom/Mask First x and Last y Characters
VARCHAR	Letters change to x, and digits change to n.	The last four characters are retained (Chinese characters remain unchanged with each character occupying one digit), and letters change to X.	The first four characters are retained (Chinese characters remain unchanged with each character occupying one digit), and letters change to X.	The value changes to its hash value of 64 bytes.	Chinese characters remain unchanged with each character occupying one digit, and letters change to X.	The value changes to null.	Encryption using the SM3 algorithm	Shows the first x characters and the last y characters.	Hides the first x characters and the last y characters.

Data Type	Masking Rule								
	Mask Letters and Digits	Show Only the Last Four Characters	Show Only the First Four Characters	Replace a Value with Its Hash Value	Mask the Month and Date	Replace a Value with Null	SM3	Custom/Show First x and Last y Characters	Custom/Mask First x and Last y Characters
DATE	The date changes to 0001-01-01.	The date changes to 0001-01-01.	The date changes to 0001-01-01.	The value changes to null.	The year is retained, and other values change to 01.	The value changes to null.	Not supported	Not supported	Not supported

Table 9-41 GaussDB(DWS) dynamic masking rules

Data Type	Masking Rule			
	Replace All Characters with Asterisks (*)	Retain Last Four Characters and Replace Others with Asterisks (*)	Retain First Two Characters and Replace Others with Asterisks (*)	Custom
Character bpchar, varchar, text, inet, macaddr, uuid, char, txt	All characters are replaced by null.	The last four characters are retained, and the other characters are replaced with asterisks (*).	The first two characters are retained, and the other characters are replaced with asterisks (*).	The start and end positions, as well as masking characters are customized.

Data Type	Masking Rule			
	Replace All Characters with Asterisks (*)	Retain Last Four Characters and Replace Others with Asterisks (*)	Retain First Two Characters and Replace Others with Asterisks (*)	Custom
Value numeric, int2, int8, money, float8, float4, interval, decimal, double precision, real, integer, smallint, bigint	All characters are replaced by 0.	Not supported	Not supported	The start and end positions, as well as masking characters are customized.
Time timestamp, time, timetz, timestamptz, date, time without time zone, timestamp without time zone, time without time zone, timestamp without time zone	All characters are replaced by a fixed value.	Not supported	Not supported	The year, month, or day can be masked as needed.
Other	All characters are replaced by a fixed value.	Not supported	Not supported	Not supported

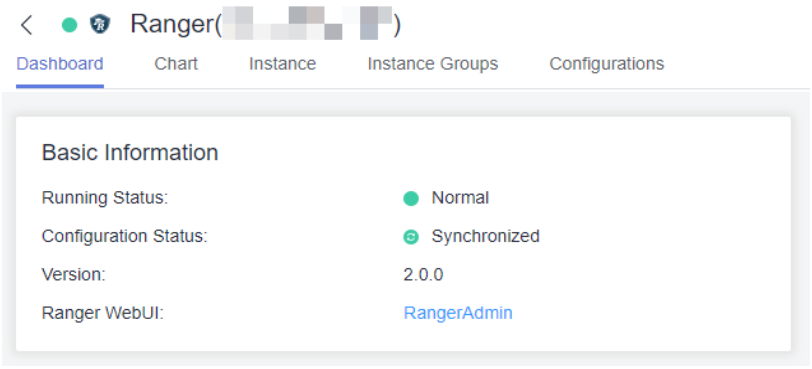
Reference: Configuring UDF-related Permissions in the Ranger Component

If you choose the SM3, Custom/Show First x and Last y Characters, or Custom/Mask First x and Last y Characters masking rule when configuring a dynamic masking policy for MRS Hive data, you must upload the JAR package on which the algorithm depends to the MRS cluster, and grant the UDF creation permission to the account in the Ranger data connection and the UDF usage permission to all users in advance. The procedure is as follows:

Step 1 Log in to MRS Manager as user **admin**.

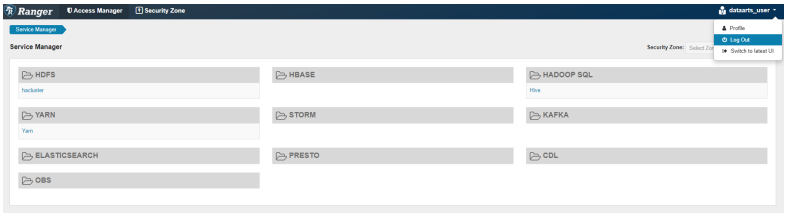
Step 2 On the Manager page, choose **Cluster** > **Services** > **Ranger**. On the Ranger overview page, click **RangerAdmin** to go to the Ranger WebUI.

Figure 9-195 Accessing the Ranger WebUI



Step 3 Log out of the current account and use the Ranger administrator account to log in again. For a common cluster, the admin account for the Manager page can be used as the Ranger administrator account. For a security cluster, **rangeradmin** is the Ranger administrator account. For details about the default password of **rangeradmin**, see **User Account List**.

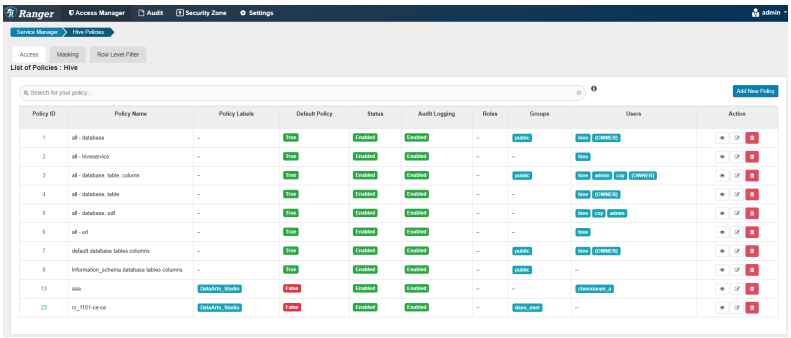
Figure 9-196 Logging out of the current account



Step 4 On the home page, click the component plug-in name in the **HADOOP SQL** area, for example, **Hive**.

Step 5 On the **Access** tab page, click **Add New Policy**.

Figure 9-197 Policy list

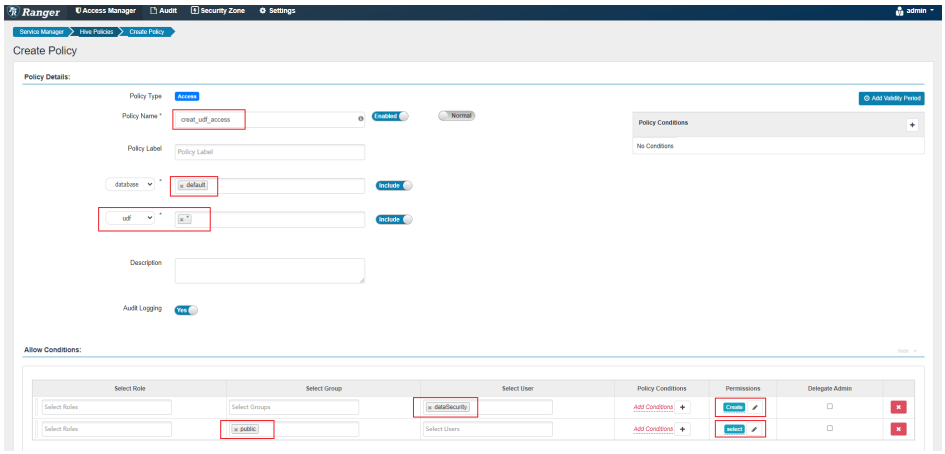


Step 6 On the **Create Policy** page, configure the policy parameters to grant the UDF creation permission to the account in the Ranger data connection and the UDF usage permission to all users.

- **Policy Name**: Enter a policy name, for example, **creat_udf_access**.
- **database**: Select **default**. The created UDF is stored in the default database by default.

- **udf**: Select * to match all UDFs.
- **Permission 1 – UDF creation permission for the account in the Ranger data connection:**
  - **Select User**: Select the account in the Ranger data connection.
  - **Permissions**: Select **Create**, which indicates the UDF creation permission.
- **Permission 2 – UDF usage permission for all users:**
  - **Select Group**: Select **public**, which indicates all users.
  - **Permissions**: Select **select**, which indicates the UDF usage permission.

Figure 9-198 Creating a policy



Step 7 Click **add** to finish configuring the permissions.

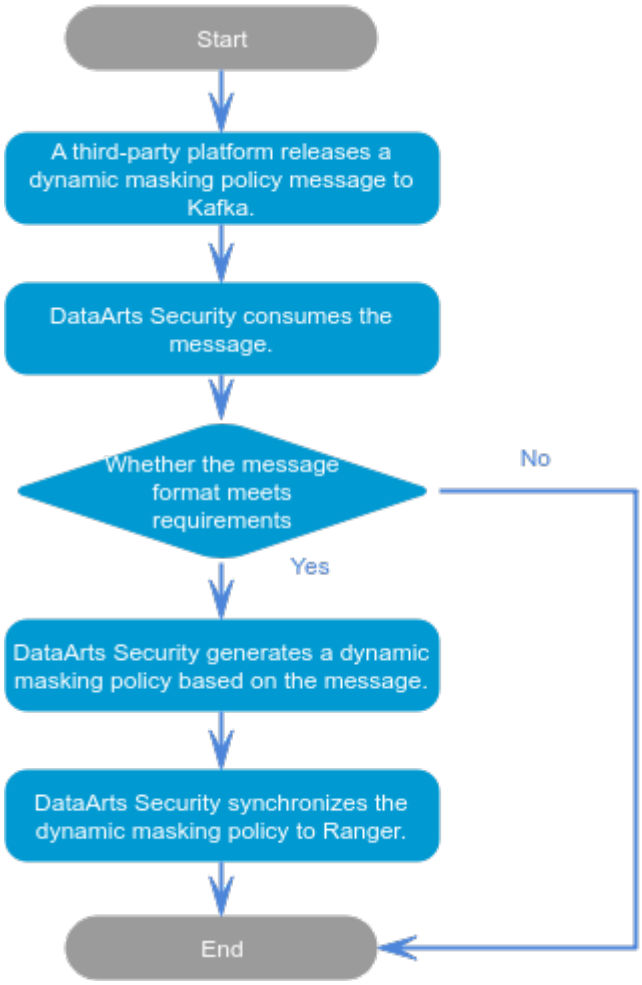
----End

9.5.3.2 Subscribing to Dynamic Masking Policies

You can synchronize dynamic masking policies from third-party platforms by subscribing to the policies.

After dynamic masking policies of third-party platforms are released to Kafka message queues, you can subscribe to and consume them in DataArts Security. If the message format meets requirements, DataArts Security generates a dynamic masking policy (whose name is the policy name in the Kafka message) and synchronizes the policy to the MRS Ranger component to make the policy take effect.

Figure 9-199 Dynamic masking policy subscription process



Note that dynamic masking subscriptions configured for a DataArts Studio instance are visible to and take effect for all the workspaces of the instance.

Prerequisites

- A dynamic masking policy of a third-party platform has been released to the Kafka message queue, and the message format meets requirements. For details, see **Reference: Kafka Message Format Requirements**.
- An MRS Kafka data connection has been created in Management Center. For details, see **Creating a Data Connection**. The Kafka must be the Kafka where the third-party platform releases a message. The account in the data connection must have the permissions of the **kafkaadmin** user group.

Constraints

- Only the DAYU Administrator, Tenant Administrator, or data security administrator can create, edit, start, stop, or synchronize dynamic masking subscription tasks. Other common users do not have permission to perform these operations.
- You can only subscribe to the dynamic masking policies for MRS Hive on third-party platforms. The dynamic masking policies support only the masking

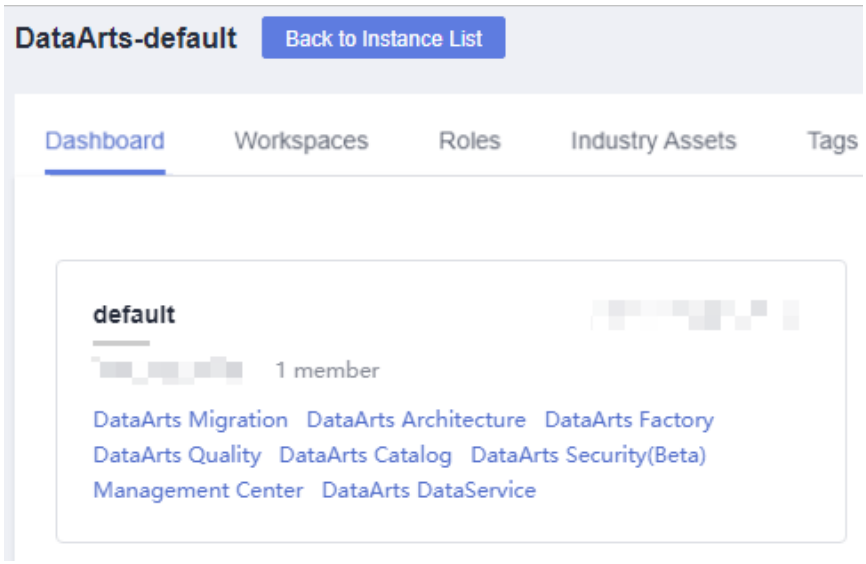
rules supported by DataArts Security. The following rules are not supported: Custom/Show First x and Last y Characters and Custom/Mask First x and Last y Characters. For details, see **Table 9-40**.

- The name of the dynamic masking policy generated by the subscription is the policy name in the Kafka message. DataArts Security does not allow duplicate policy names. Ensure that no dynamic masking policy name is the same as any policy name in the Kafka message.
- After the dynamic masking policy generated by the subscription is synchronized to Ranger, the policy name is **dlsMasking-***Database name*-*Table name*-*Column name*. Ranger does not allow duplicate policy names. Ensure that no existing policy name in Ranger is the same as the name of any generated policy.
- During dynamic masking subscription, DataArts Security uses the MRS cluster in the subscription task and the database, table, and column in the Kafka message dynamic masking policy to identify a dynamic masking policy. If a dynamic masking policy for the same table column in the same cluster's database already exists in the message queue or DataArts Security, the policy is skipped and will not be generated.
- The SM3, Custom/Show First x and Last y Characters, and Custom/Mask First x and Last y Characters masking rules for the MRS Hive data source are not provided by the MRS Ranger component. Instead, they are implemented by UDF-defined functions. Therefore, to use any of the three masking rules, you must upload the JAR package on which the algorithm depends to the MRS cluster, and grant the UDF creation permission to the account in the Ranger data connection and the UDF usage permission to all users in advance. For details, see **Reference: Configuring UDF-related Permissions in the Ranger Component**.
- DataArts Security can consume a Kafka message only if the message format meets the requirements described in **Reference: Kafka Message Format Requirements**.
  - If the Kafka message does not meet the message format requirements, the system records a synchronization failure message log and continues to consume the next message. The final status is partially failed or synchronization failed.
  - If the Kafka message is valid but fails to be consumed due to network resource issues, the consumption will be retried three times at intervals of 4, 6, and 9 seconds. If the message still fails to be consumed, a log will be recorded and the scheduling will be terminated.
  - If the Kafka message is valid and consumed properly, but a policy fails to be generated or synchronized to Ranger, the system records a synchronization failure message log and continues to consume the next message. The final status is partially failed or synchronization failed.
  - A maximum of 16 MB of failed Kafka messages can be stored.

Subscribing to Dynamic Masking Policies

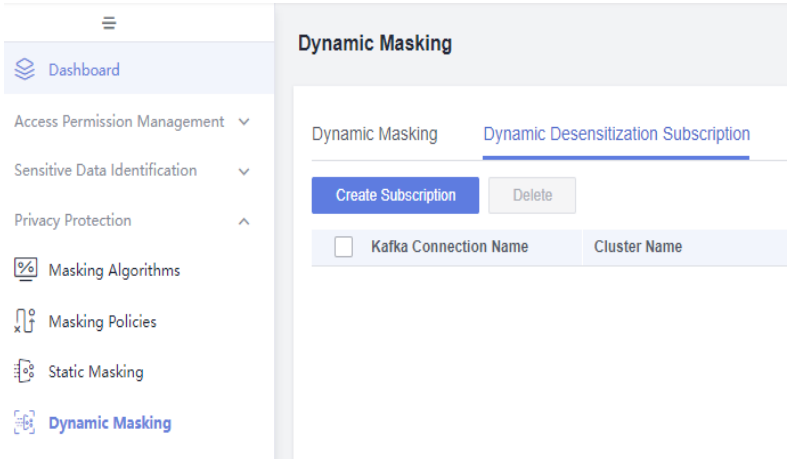
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-200 DataArts Security



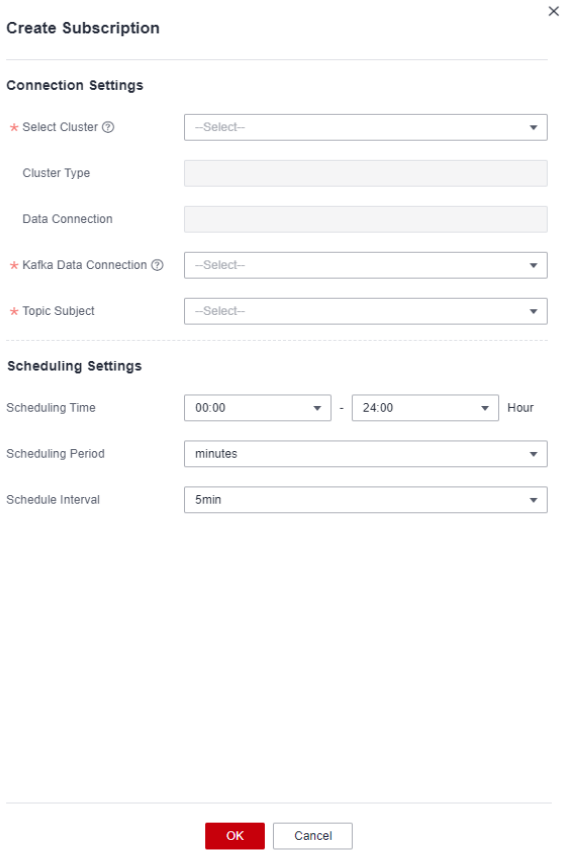
Step 2 In the navigation pane on the left, choose **Dynamic Masking**. On the displayed page, click the **Dynamic Desensitization Subscription** tab.

Figure 9-201 Dynamic Desensitization Subscription tab



Step 3 Click **Create Subscription**. In the displayed slide-out panel, set the parameters listed in **Table 9-42**.

Figure 9-202 Parameters for creating a subscription



The following table lists the parameters for creating a dynamic masking subscription.

Table 9-42 Parameters

Parameter	Description
Connection Settings	
*Select Cluster	Select the cluster to which a dynamic masking policy of a third-party platform will be synchronized. Currently, a policy cannot be synchronized to multiple clusters. If you want to do so by creating multiple subscription tasks, Kafka messages will fail to be consumed due to duplicate policy names.
Cluster Type	You do not need to set this parameter. The system automatically sets it based on the cluster you select. Currently, policies can only be synchronized to an MRS cluster.
Data Connection	You do not need to set this parameter. The system automatically sets it based on the cluster you select.

Parameter	Description
*Kafka Data Connection	Select the MRS Kafka connection created in **Prerequisites**. The Kafka must be the Kafka where the third-party platform releases a message. The account in the Kafka connection must have the permissions of the **kafkaadmin** user group.
*Topic Subject	Select the topic of the Kafka message released for the dynamic masking policy of the third-party platform. A topic in the same MRS cluster can correspond to only one subscription task.
Scheduling Settings	
Scheduling Time	Select the time period every day during which tasks will be scheduled. Set an appropriate time period based on the number of messages. Currently, it takes about two seconds to consume and synchronize a piece of data.
Scheduling Period	Set whether to schedule tasks by hour or minute.
Schedule Interval	Select the interval at which tasks are shceduled.

Step 4 After setting all required parameters, click **OK**. Then click **Start** to start task scheduling.

----End

Related Operations

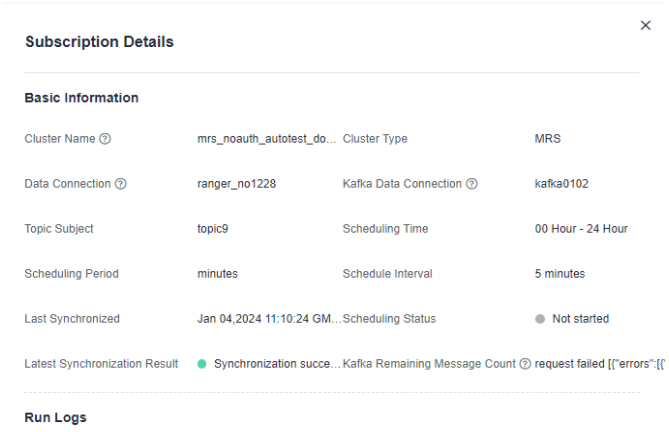
- Starting or stopping a subscription task: On the **Dynamic Desensitization Subscription** tab page, locate a subscription task and click **Start** or **Stop** in the **Operation** column.
- Editing a subscription task: On the **Dynamic Desensitization Subscription** tab page, locate a subscription task, click **More** in the **Operation** column, and select **Edit**.
- Deleting subscription tasks: On the **Dynamic Desensitization Subscription** tab page, locate a subscription task, click **More** in the **Operation** column, and select **Delete**. To delete multiple tasks, select them and click **Delete** above the task list.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.
- Synchronizing a subscription task: On the **Dynamic Desensitization Subscription** tab page, locate a subscription task, click **More** in the **Operation** column, and select **Synchronize**. After that, DataArts Security consumes the message, generates a policy, and synchronizes the policy to Ranger.

- Viewing subscription task details: On the **Dynamic Desensitization Subscription** tab page, locate a task, and click **Details** in the **Operation** column to view the task details.

  Figure 9-203 Viewing task details



Reference: Kafka Message Format Requirements

Dynamic masking policies of third-party platforms need to be released to a Kafka message queue, and the message format must meet requirements. The following is a message template with parameters.

```
{
  "mask_policy_template":
  {
      "create_time":1692839884000  //Synchronization time
      "name":" task1", //Name of the dynamic masking policy, which cannot be the same as the name of any existing dynamic masking policy
      "database": "1", //Database name
      "table": "1", //Data table name
      "column": "1", //Field name
      "column_type":"int", //Field type
      "data_level": "1", //Field security level, which is optional
      "algorithm_config": {
        "name": "SM3", //Dynamic masking rule name, which can be MASK, MASK_SHOW_LAST_4, MASK_SHOW_FIRST_4, MASK_HASH, MASK_DATE_SHOW_YEAR, MASK_NULL, or SM3
        "type": "HASH", ||Dynamic masking rule type, which is MASK for all rules except the SM3 rule whose type is HASH
        "description": "Encryption using the SM3 algorithm", //Dynamic masking rule description
      },
      "datasource_type":"HIVE", //Data source type, which can only be Hive
      "users":"aaa,bbb",  //Masking users
      "user_groups":"ggg"  //Masking user groups
      "description":{
          "jdbc_url": "hive2://xxx" //Custom description, which is contained in a failure message
      }
  }
}
```

9.5.4 Managing Data Watermarks

9.5.4.1 Embedding Data Watermarks

This section describes how to embed data watermarks. Data watermarking applies to the following scenarios:

- Data forwarding process standardization

  Unauthorized users need to be approved when they forward data from enterprises for external usage. After the approval, the watermark technology is used to generate files that can be used outside enterprises.
- Digital right protection

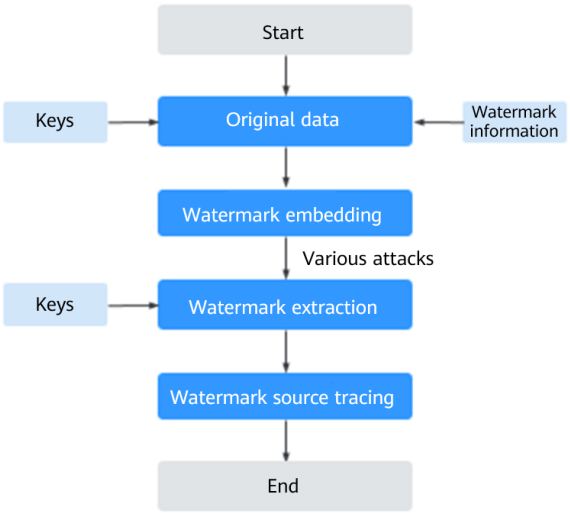
  To associate databases with their owners, embed watermarks representing ownerships in relational databases. In this way, enterprises' digital rights can be protected.
- Quick source tracing of leaked data

  To locate security vulnerabilities, unseal the leaked data files, check whether watermarks exist based on the file integrity and watermark traces, and identify watermark information such as data source addresses, distribution units, owners, and distribution time.

Watermark Use Process

Figure 9-204 shows the process of using watermarks.

Figure 9-204 Watermark use process



Constraints

- Only the MRS Hive data sources support data watermark tasks.
- Watermarks cannot be embedded into a primary key.
- If a watermark is embedded in a numeric integer field, the data may be modified. Embed watermarks into a field whose value can be changed.
- If **Dataset Scope** is set to **Incremental** for a data watermark embedding task, **Timestamp** or **Date** needs to be selected for **Time Field**.

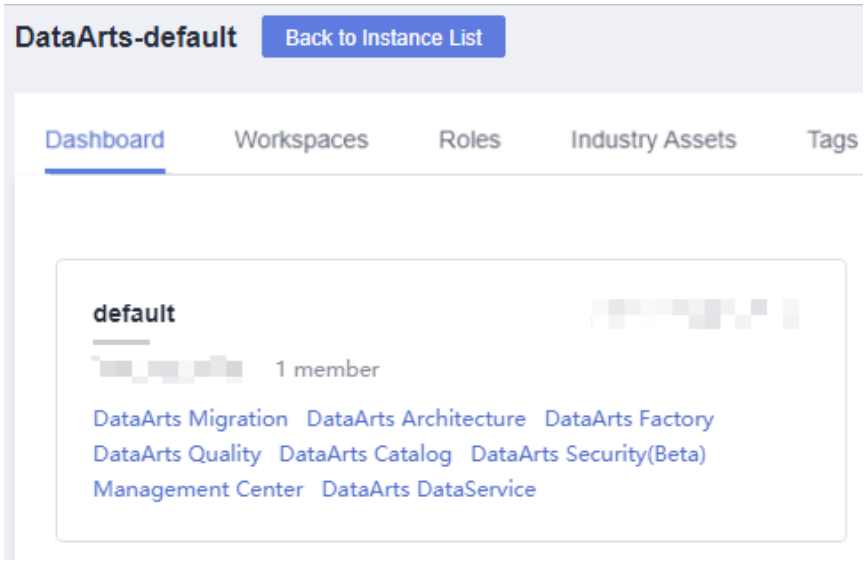
Prerequisites

An MRS Hive connection has been created. For details, see **Creating a Data Connection**.

Creating a Data Watermark Embedding Task

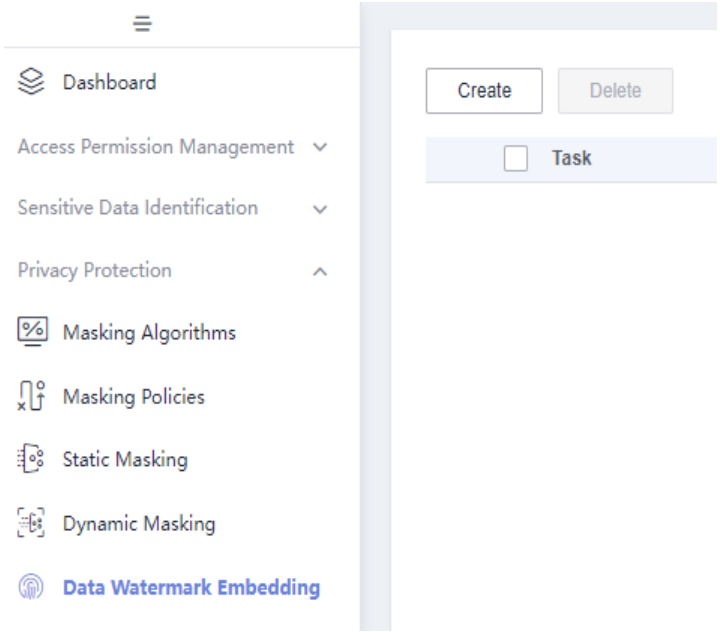
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-205 DataArts Security



Step 2 Choose **Data Watermark Embedding** from the left navigation bar, and click **Create** in the upper part of the displayed page.

Figure 9-206 Creating a data watermark embedding task

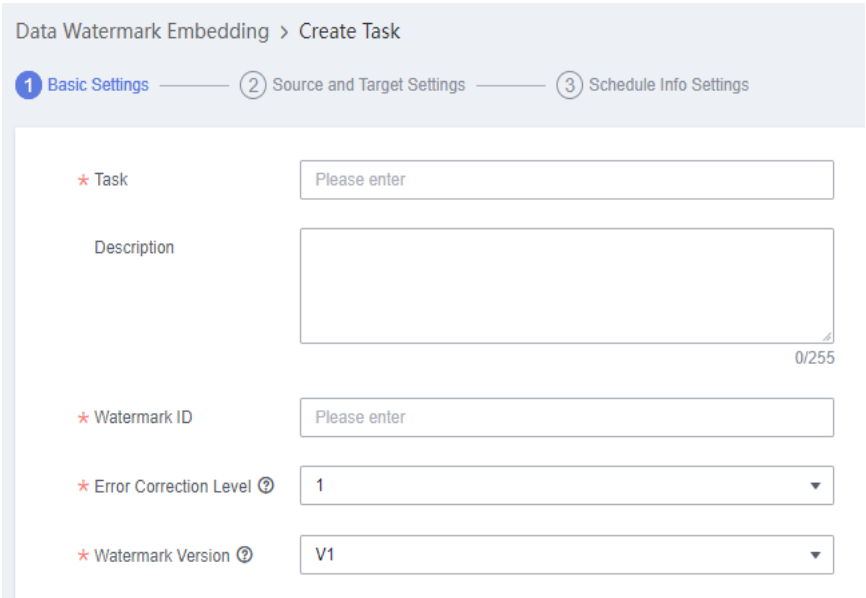


Step 3 In the displayed dialog box, set the parameters listed in **Table 9-43**.

Table 9-43 Basic settings

Parameter	Description
*Task	Name of the watermark embedding task. The name can only contain letters, digits, underscores (_), and hyphens (-), and can contain a maximum of 64 characters. To facilitate the management of the watermark embedding task, you are advised to include the object into which you want to embed the watermark and the watermark ID in the name.
Description	A description of the task
*Watermark ID	Watermark ID that will be embedded by the system into data tables. The watermark ID can contain a maximum of 16 characters.
*Error Correction Level	The higher the level, the more bits of watermark information, and the lower bit error rate during source tracing. Note that a higher error correction level requires a larger amount of data to ensure the integrity of embedded information. The default value is **1**.
*Watermark Version	**V1**: Watermarks depend on primary keys, and the embedding speed is fast. If primary keys are attacked, source tracing may fail. **V2**: Watermarks do not depend on primary keys. They are related only to embed columns. The embedding speed is slow and the robustness is enhanced.

Figure 9-207 Configuring basic information

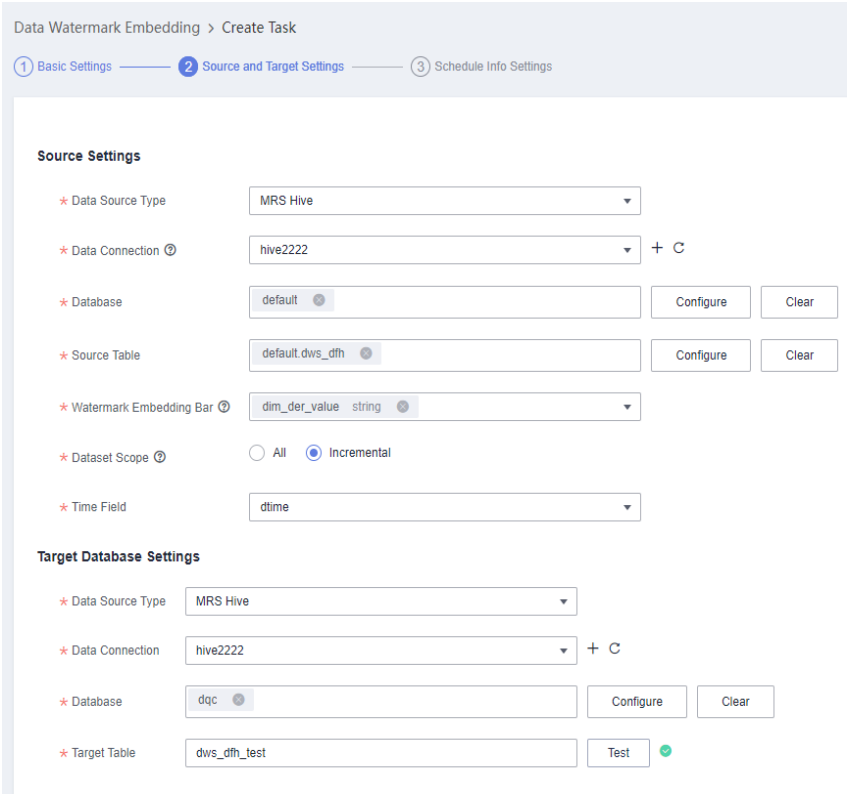


Step 4 Click **Next** to configure the source and target end parameters listed in **Table 9-44**.

Table 9-44 Source and target end parameters

Parameter	Description
Source Settings	
*Data Source Type	Currently, the value can only be **MRS Hive**.
*Data Connection	Select a data connection. If no data connection is available, create one by referring to **Creating a Data Connection**.
*Database	Select the databases and tables into which you want to embed the watermark. • Click **Configure** to select databases and tables. • Click **Clear** to delete the selected databases and data tables.
*Source Table	
*Watermark Embedding Bar	Select a field type from the drop-down list as the embedding bar. For example, the value can be a number or a character. Note that when **Watermark Version** is set to **V1**, the primary key column cannot be selected.
*Dataset Scope	If **Dataset Scope** is set to **Incremental**, you can set **Time Field** to **Timestamp** or **Date**. Generally, the watermark embedding task is scheduled once if this parameter is set to **All** and is scheduled periodically if this parameter is set to **Incremental**.
*Time Field	If **Dataset Scope** is set to **Incremental**, you can set this parameter to **Timestamp** or **Date**.
Target End Settings	
*Data Source Type	Currently, the value can only be **MRS Hive**.
*Data Connection	Select a data connection. If no data connection is available, create one by referring to **Creating a Data Connection**.
*Database	Select the database where the watermark table is stored from the drop-down list.
*Target Table	Enter a unique table name. The table is automatically created when the table name entered does not exist. Click **Test**. Otherwise, the next operation is not allowed.

Figure 9-208 Configuring source and target information



Step 5 Click **Next** and configure scheduling.

- If **Dataset Scope** is set to **All**, **Repeat** can be only set to **Once**.
- If **Dataset Scope** is set to **Incremental**, **Repeat** can be set to **Once** or **On Schedule**.

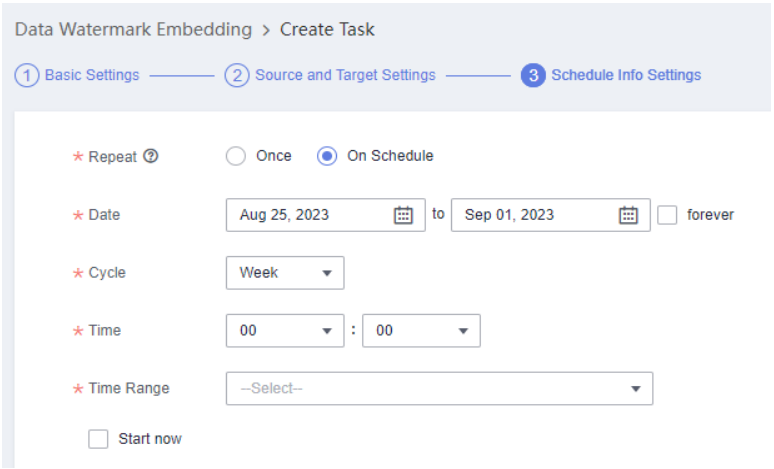
If you set **Repeat** to **On Schedule**, set the parameters listed in **Table 9-45**.

Table 9-45 Parameters for periodic scheduling

Parameter	Description
*Date	Period during which the task takes effect.

Parameter	Description
*Cycle	The frequency at which a task is executed. The options are: • **minutes**: Select the scheduling start time and end time, and set the interval in minutes. • **hours**: Select the scheduling start time and end time, and set the interval in hours. • **Day**: Set the scheduling time everyday. • **Week**: Select a day in a week and set the specific time to start scheduling. • **Month**: Select a day in a month and set the specific time to start scheduling. For example, you can set **Cycle** to **Week**, **Time** to **15:52**, and **Time Range** to **Tuesday**. In this case, the task is executed at 15:52 every Tuesday within the configured date range.
Start now	If you select **Start now**, the task is scheduled immediately.

Figure 9-209 Configuring scheduling



Step 6 Click **OK**.

----End

Related Operations

- Editing a task: On the **Data Watermark Embedding** page, locate a task and click **Edit** in the **Operation** column.

  A task in the **Scheduling** state cannot be edited.
- Deleting tasks: On the **Data Watermark Embedding** page, locate a task, click **More** in the **Operation** column, and select **Delete**. To delete multiple tasks at a time, select the tasks and click **Delete** above the task list.

A task in the **Scheduling** state cannot be deleted.

NOTE

The deletion operation cannot be undone. Exercise caution when performing this operation.

- Running or scheduling a task: On the **Data Watermark Embedding** page, locate a task and click **Run** in the **Operation** column or click **More** in the **Operation** column and select **Start**.

  You can determine whether a task is scheduled once or repeatedly based on the scheduling period.
- Viewing running instance logs: On the **Data Watermark Embedding** page, locate a task and click to expand instances. Then click **View Log**.

  If a task fails to be executed, you can locate the failure cause based on logs, rectify the fault, and try the task again. If the fault persists, contact technical support.

9.5.4.2 Tracing Data Using Watermarks

This section describes how to use watermarks to trace leaked data in files.

DataArts Security provides users with the source tracing function to accurately trace the leaked data. Users can check whether watermarks exist based on the leaked data file integrity and watermark traces, identify watermark traces, and accurately locate the security issues and find the personnel or departments accountable for the leakage problem.

Prerequisites

- After obtaining the leaked data file, a CSV (Comma-Separated Values) file whose size does not exceed 20 MB has been generated and saved to the local host.
- A data watermark embedding task has been created. For details, see **Embedding Data Watermarks**.

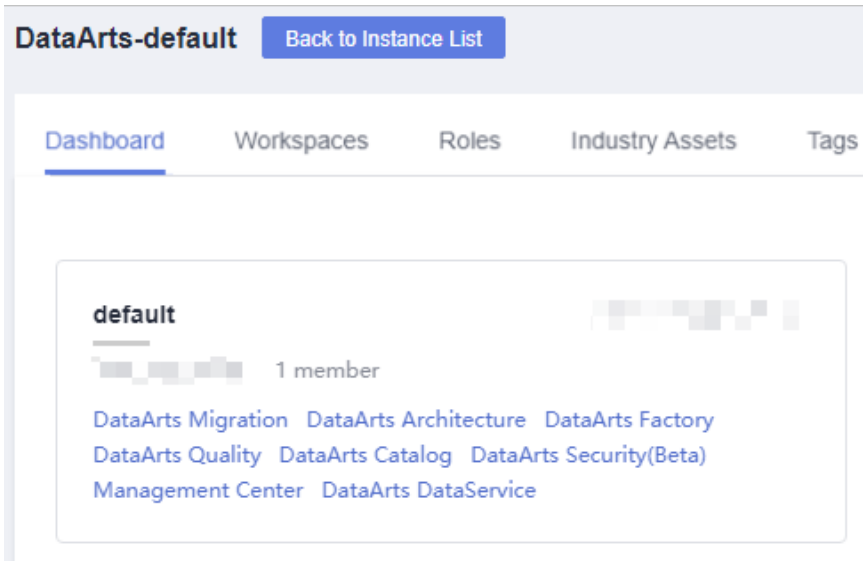
Notes and Constraints

- Watermarks can only be used to trace data in files no larger than 20 MB.
- To trace data accurately, ensure the integrity and correctness of the data. The first column of the target table data file cannot be empty, and the file should contain more than 5,000 data records.

Creating a Data Watermark Source Tracing Task

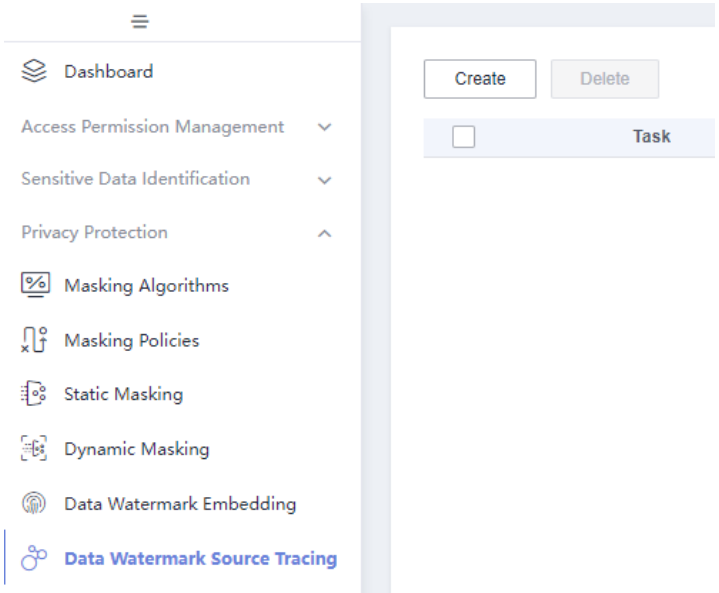
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-210 DataArts Security



Step 2 Choose **Data Watermark Source Tracing** from the left navigation bar, and click **Create** in the upper part of the displayed page.

Figure 9-211 Creating a source tracing task



Step 3 In the displayed dialog box, set the parameters listed in **Table 9-46**.

Figure 9-212 Creating a source tracing task

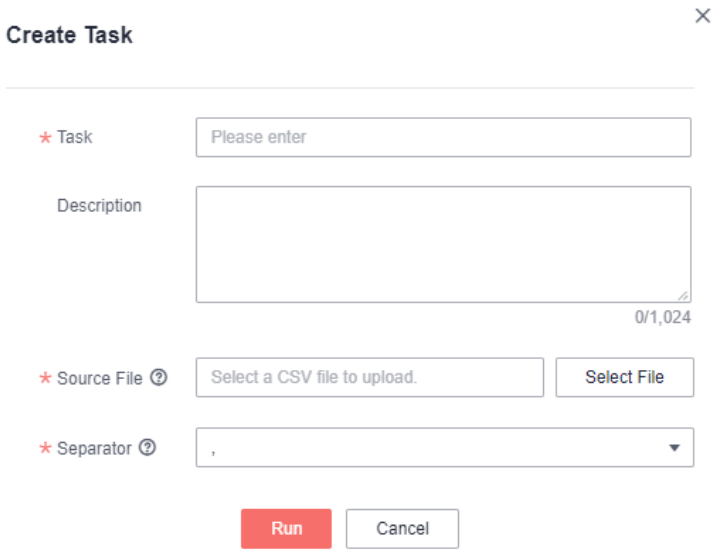


Table 9-46 Parameters

Parameter	Description
Task	The name of the watermark task to be created. Task names can include only letters, numbers, underscores (_), and hyphens (-), and cannot exceed 64 characters.
Description	A description of the task. The description can contain a maximum of 1,024 characters.
Source File	CSV file generated from the leaked data file. The file cannot be larger than 20 MB.
Separator	Select a separator from the drop-down list based on the uploaded CSV file. The options are **Comma (,)**, **Tab**, **Vertical bar (|)**, and **Semicolon (;)**. By default, **Comma (,)** is selected.

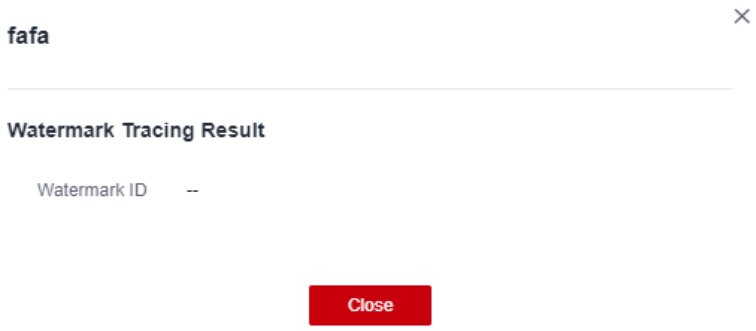
Step 4 After all settings are complete, click **Run**.

----End

Related Operations

- Viewing the source tracing result: On the **Data Watermark Source Tracing** page, locate a task and click **View Result** in the **Operation** column. Source tracing results are displayed only for the tasks that have been successfully executed.

Figure 9-213 Source tracing result



- Deleting tasks: On the **Data Watermark Source Tracing** page, locate a task and click **Delete** in the **Operation** column. To delete multiple tasks at a time, select the tasks and click **Delete** above the task list.

  A task in the **Scheduling** state cannot be deleted.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.

9.5.5 Managing File Watermarks

This section describes the following operations on file watermarks:

- Insert invisible watermarks into structured data files (CSV, XML, and JSON files) and extract the watermarks.
- Insert visible watermarks into unstructured data files (DOCX, PPTX, XLSX, and PDF files) and open the files on a local host to view the watermarks.

Constraints

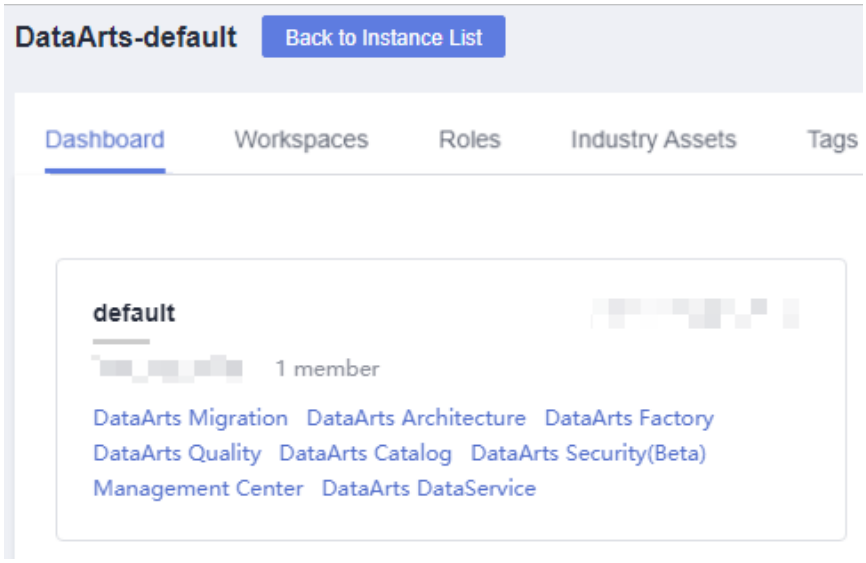
- Invisible watermarks can be inserted into and extracted from structured data files that are no longer than 4 MB.
- Visible watermarks can be inserted into unstructured data files that are no longer than 20 MB.
- Watermarks cannot be injected into files that already contain watermarks.
- The data in structured data files into which watermarks are to be inserted must meet the following requirements:
  - The source data must contain 5,000 or more lines. If the source data contains less than 5,000 lines, watermarks may fail to be extracted due to insufficient features.
  - You are advised to select a column with various data values. If all the values of the column can be enumerated, the extraction may fail due to insufficient features. Common columns that can be embedded with watermarks include the address, name, UUID, amount, and total amount.
  - If a watermark is inserted into a numeric integer field, the data may be modified. Insert watermarks into a field whose value can be changed.
- Watermark extraction from structured data files is irrelevant to the source tracing tasks using data watermarks. Only users under the same account can

extract watermarks from structured data files into which watermarks have been inserted by following the instructions in **Inserting a Watermark** or **Managing Dynamic Watermarking Policies**.

Inserting a Watermark

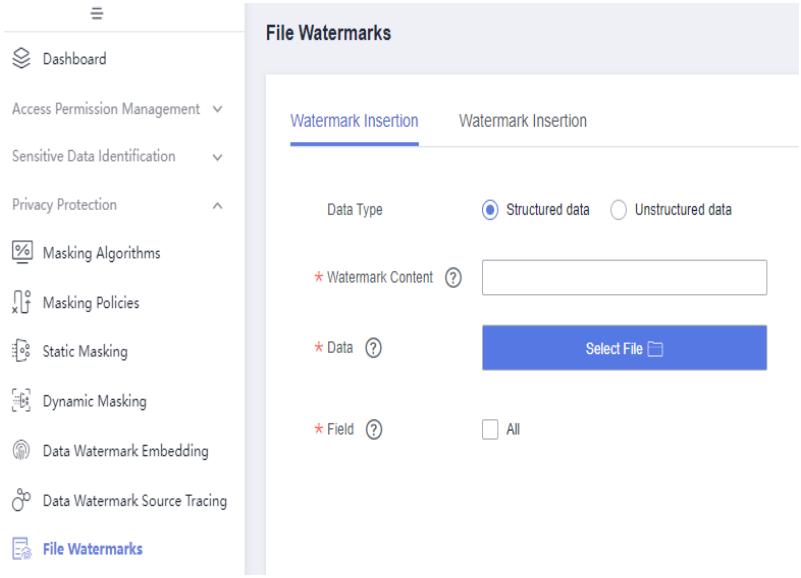
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-214 DataArts Security



Step 2 In the left navigation pane, choose **File Watermarks**.

Figure 9-215 Accessing the File Watermarks page



Step 3 Set the parameters listed in **Table 9-47**.

Table 9-47 Parameters for inserting a watermark

Parameter	Description
*Data Type	Select a file type. • **Structured data**: CSV, XML, and JSON. You can insert an invisible watermark into a file and extract the watermark. • **Unstructured data**: DOCX, PPTX, XLSX, and PDF You can insert a visible watermark into a file and open the file to view the watermark.
Structured data	
*Watermark Content	Watermark ID that will be embedded by the system into data tables. The watermark ID can contain a maximum of 16 characters.
*Data	CSV, XML, or JSON files are supported.
*Field	Fields into which the watermark is to be inserted.
Unstructured data	
*Watermark Content	Watermark ID that will be embedded by the system into data tables. The watermark ID can contain a maximum of 16 characters.
Transparency	Transparency of the plaintext watermark
Rotation Angle	Rotation angle of the plaintext watermark
Font Size	Font size of the plaintext watermark
*Data	DOCX, PPTX, XLSX, and PDF files are supported.

Step 4 Click **Insert Watermark**. The browser automatically downloads the inserted file.

You can click **Reset** to restore the parameters to default settings.

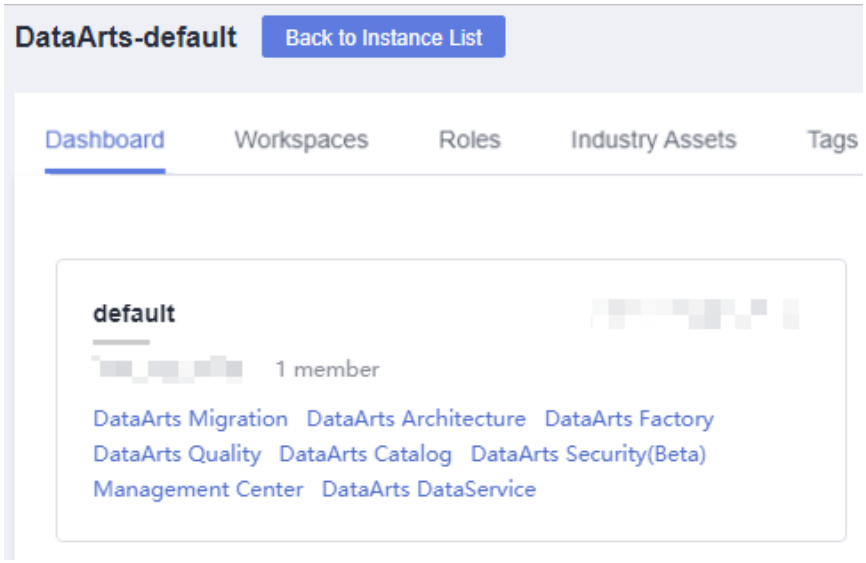
----End

Extracting a Watermark

You can extract invisible watermarks that have been inserted into structured data files in CSV, XML, or JSON format. For details about watermark insertion, see **Inserting a Watermark**.

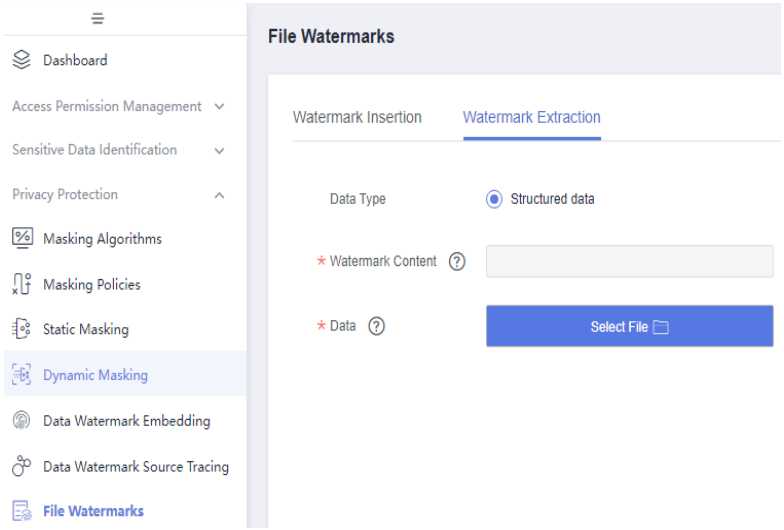
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-216 DataArts Security



Step 2 In the left navigation pane, choose **File Watermarks**. In the right pane, click the **Watermark Extraction** tab.

Figure 9-217 Accessing the Watermark Extraction page



Step 3 Set the parameters listed in **Table 9-48**.

Table 9-48 Parameters for extracting a watermark

Parameter	Description
*Data Type	File type. Only CSV, XML, and JSON are supported. You can insert an invisible watermark into a file of any preceding type and extract the watermark.

Parameter	Description
*Watermark Content	You do not need to set this parameter. The extracted watermark will be automatically displayed.
*Data	Select the structured data file in CSV, XML, or JSON format into which an invisible watermark has been inserted based on **Inserting a Watermark**.

Step 4 Click **Extract Watermark**. The extracted watermark is displayed in the **Watermark Content** parameter.

You can click **Reset** to restore the parameters to default settings.

----End

9.5.6 Managing Dynamic Watermarking Policies

Dynamic watermarking means dynamically inserting watermarks into the result sets returned by data query and access requests. This section describes how to enable dynamic watermarking for DataArts Factory so that data watermarks can be dynamically inserted during the dump or download of sensitive data in DataArts Factory.

After dynamic watermarking is enabled for DataArts Factory and a dynamic watermarking policy is created in DataArts Security, an invisible dark watermark will be inserted into the sensitive data dumped or downloaded by a user group or role specified in the policy to prevent the sensitive data from being disclosed. The watermark is the first 16 digits from the ID of the IAM user who is attempting to obtain the sensitive data. For details about how to view the IAM user ID, see "Obtaining a Project ID and Account ID" in **(Optional) Obtaining Authentication Information**.

Note that dynamic watermarking policies configured for a DataArts Studio instance are visible to and take effect for all the workspaces of the instance.

Prerequisites

- An MRS Hive or MRS Spark connection has been created.

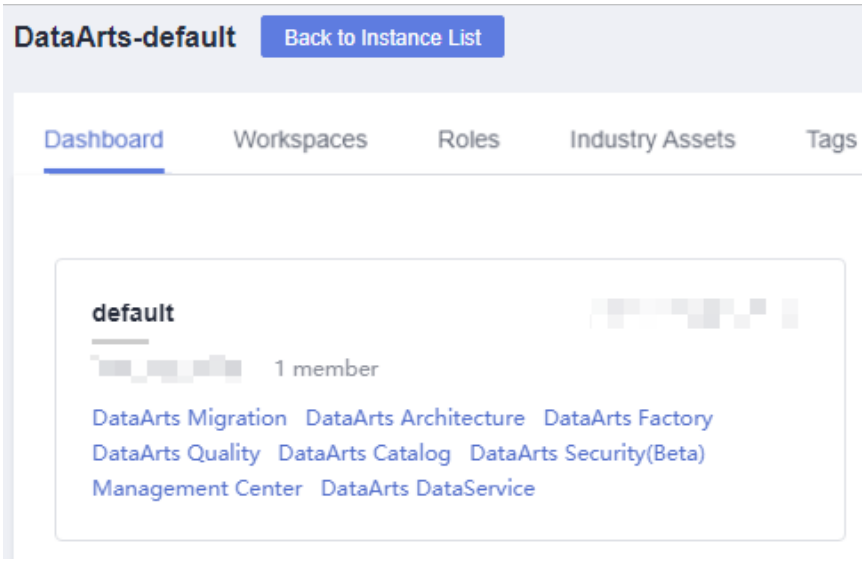
Constraints

- Only the DAYU Administrator, Tenant Administrator, or data security administrator can enable or disable dynamic watermarking for DataArts Factory. The workspace administrator can create dynamic watermarking policies. Other common users do not have the permission to perform these operations.
- Dynamic watermarking policies are only available for MRS Hive and MRS Spark data sources.
- Adding, deleting, or modifying a dynamic watermarking policy takes about five minutes to take effect.
- A watermark will be inserted only when more than 500 rows of data are to be dumped or downloaded. If there are less than 500 rows of data, source tracing will be impossible even if a watermark is inserted.

Creating a Dynamic Watermarking Policy

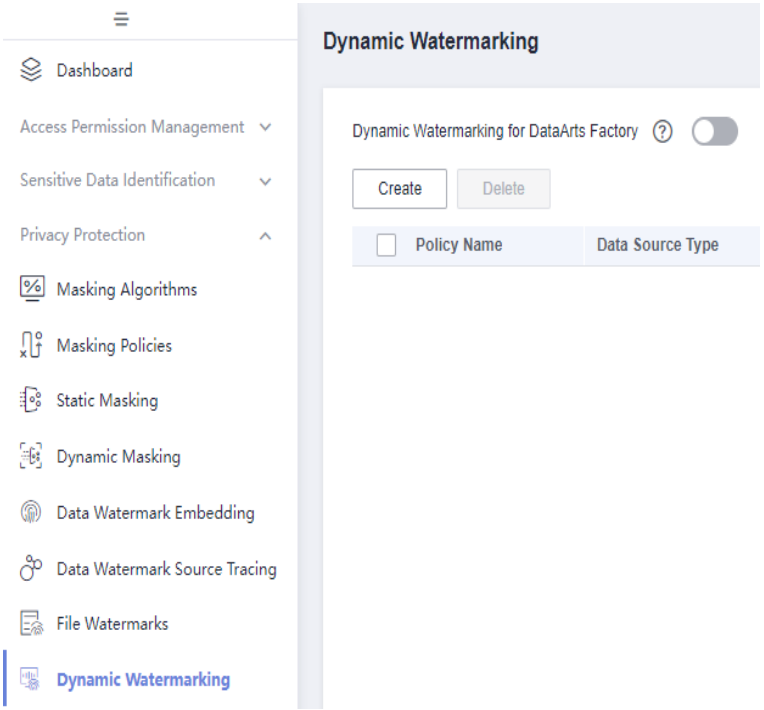
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-218 DataArts Security



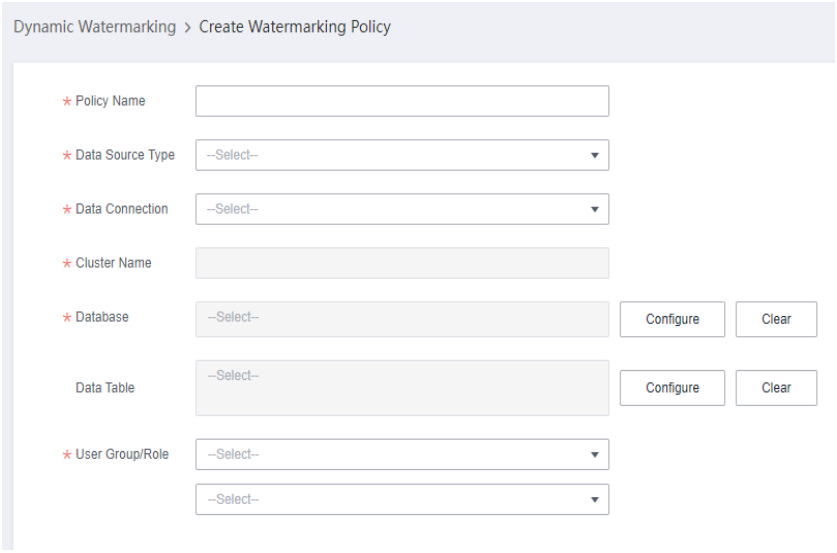
Step 2 In the left navigation pane, choose **Dynamic Watermarking**.

Figure 9-219 Accessing the Dynamic Watermarking page



Step 3 Click  to enable dynamic watermarking for DataArts Factory. Click **Create** and set the parameters listed in **Table 9-49**.

Figure 9-220 Setting parameters for the dynamic watermarking policy



The following table lists the parameters.

Table 9-49 Policy parameters

Parameter	Description
*Policy Name	Unique identifier of the dynamic watermarking policy. It must be unique in a DataArts Studio instance. To facilitate policy management, you are advised to include the object to be watermarked and the watermark to be added in the name.
*Data Source Type	Select **MRS Hive** or **MRS Spark**.
*Data Connection	If no data connection is available, create one by referring to **Creating a Data Connection**.
*Cluster Name	You do not need to set this parameter. The data source cluster in the data connection is automatically selected.
*Database	Database where the sensitive data is stored
*Data Table	Data table where the sensitive data is stored
User Group/ Role	User, user group, or role in the current workspace members. When a specified object queries or exports sensitive data from DataArts Factory, the system adds a dynamic watermark to the sensitive data to protect the sensitive data from being disclosed.

Step 4 After setting all required parameters, click **OK**.

----End

Related Operations

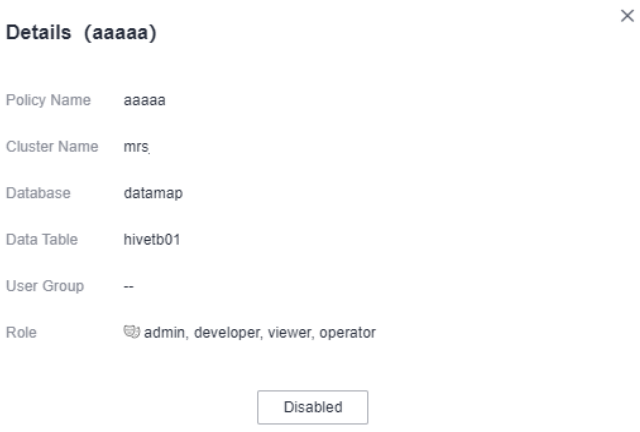
- Extracting a watermark: After obtaining the CSV data file containing a dynamic watermark from DataArts Factory, trace the watermark by referring to **Extracting a Watermark**.
- Editing a policy: On the **Dynamic Watermarking** page, locate a policy and click **Edit** in the **Operation** column.
- Setting the policy status: A watermarking policy is enabled by default. If the watermarking policy is disabled, it does not take effect.

  To change the status of a watermarking policy, click  or  to enable or disable the policy.
- Deleting policies: On the **Dynamic Watermarking** page, locate a policy and click **Delete** in the **Operation** column. To delete multiple policies, select them and click **Delete** above the list.

  NOTE

  The deletion operation cannot be undone. Exercise caution when performing this operation.
- Viewing policy details: On the **Dynamic Watermarking** page, locate a policy and click its name to view its details.

  Figure 9-221 Viewing policy details



9.6 Data Security Operations

9.6.1 Viewing Audit Logs

DataArts Security provides detailed data operation logs for GaussDB(DWS), DLI, and Hive data sources. The logs contain the time, users, objects, and types of operations. Based on these logs, you can quickly audit data operations and better manage data security.

Prerequisites

To audit access to GaussDB(DWS) data sources, ensure that the following conditions are met:

- The audit function has been enabled for GaussDB(DWS) clusters.

  The audit function is enabled by default. If it is disabled, set **audit_enabled** to **ON** by following the instructions in **Modifying Database Parameters**.
- The items to be audited have been configured.

  For details about GaussDB(DWS) audit items and how to enable them, see **Configuring the Database Audit Logs**.

Constraints

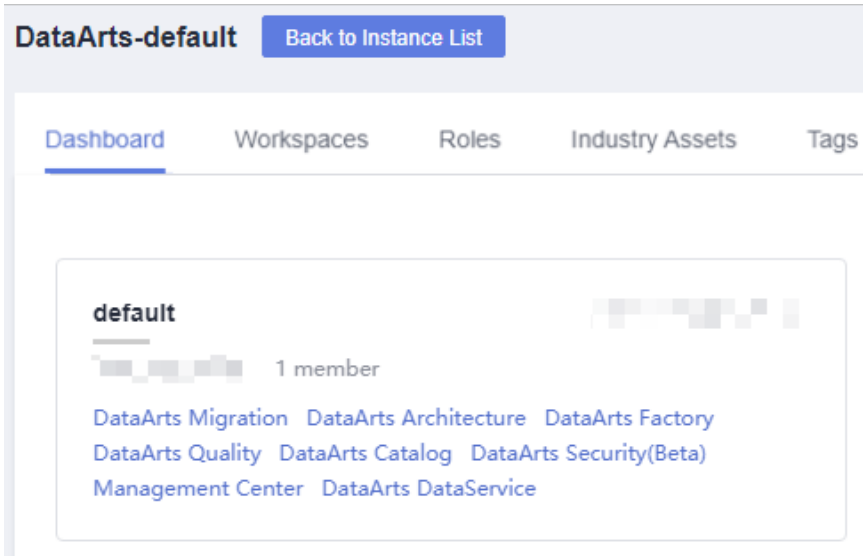
For the GaussDB(DWS) data source, if separation of duties is disabled, users with the SYSADMIN attribute can view audit records by default. If separation of duties is enabled, only users with the AUDITADMIN attribute can view audit records. Therefore, ensure that the account in the data connection or the current user has the preceding permissions. (Before enabling the permission application, use the account in the data connection to view audit records. If the permission application is enabled, use the current IAM user to view audit records.)

The MRS audit data depends on the CDM agent. Ensure that the CDM agent version suits Huawei Cloud Stack 8.3.1.

Procedure

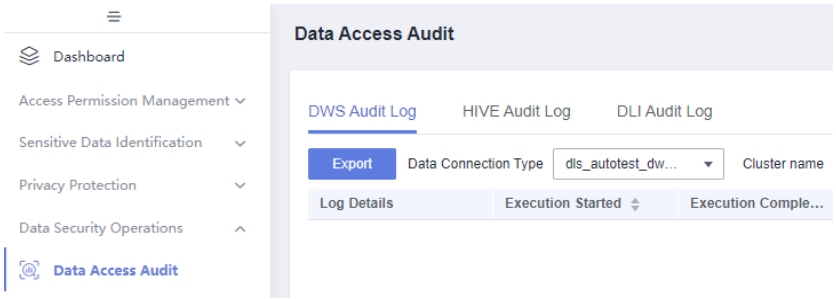
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-222 DataArts Security



Step 2 In the left navigation pane, choose **Data Access Audit**.

Figure 9-223 Data Access Audit

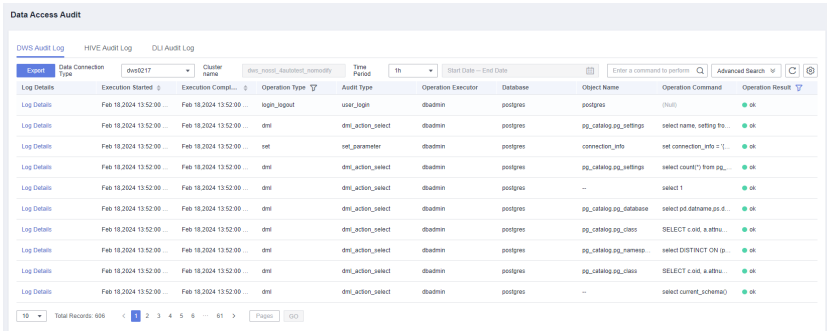


Step 3 You can switch between tabs to view the audit logs of different data sources. By default, logs generated in the last one hour are displayed. You can customize the time range, which can be up to one month.

- DWS audit log: The log list uses the latest DWS data connection by default. Click **Log Details** to view information about a log.

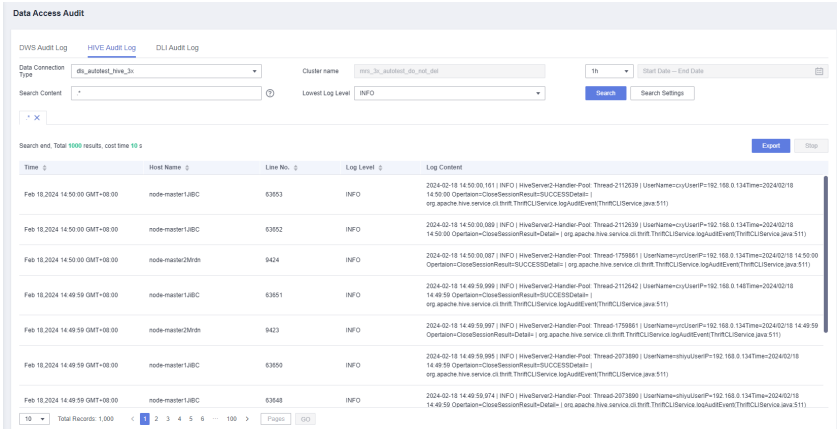
  Click **Export** to export DWS audit logs on the current page in JSON format.

  Figure 9-224 DWS audit logs

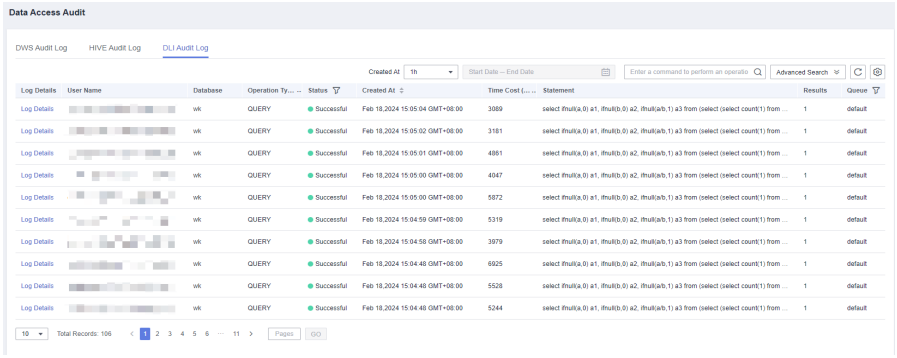
- MRS Hive audit logs: By default, the MRS Hive log list does not display log content. You can search for logs based on conditions. The search results are displayed by tab page. A maximum of five tab pages of search results can be displayed.

  Figure 9-225 MRS Hive audit logs

- DLI audit logs: By default, the DLI log list displays log information. Click **Log Details** to view information about a log.

Figure 9-226 DLI audit logs



----End

9.6.2 Diagnosing Data Security Risks

Data security diagnosis can help you diagnose data security capabilities and provide rectification suggestions and solutions for you based on the diagnosis result. In this way, you can quickly establish a basic data security system to ensure data security and reliability.

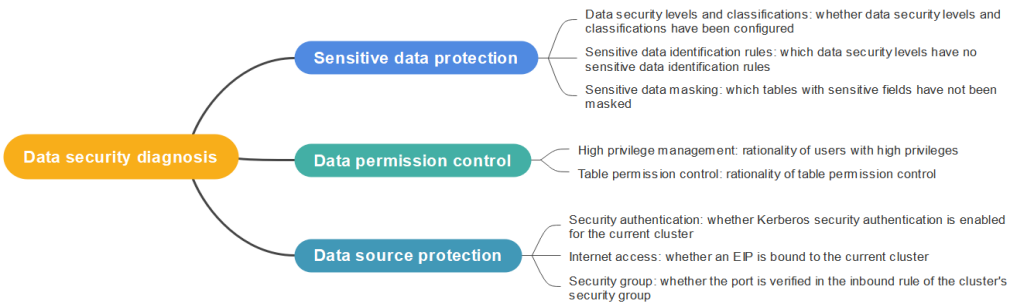
Constraints

- Currently, only the security of the MRS data source can be diagnosed.
- The timeout duration of a scanning task for security diagnosis is one hour.
- For the data permission control diagnosis item, the workspace administrator and security administrator only collect statistics of users, but not of user group members.

Diagnosing Data Security Risks

Data security diagnosis supports three diagnosis items: sensitive data protection, data permission control, and data source protection. For details, see **Figure 9-227**.

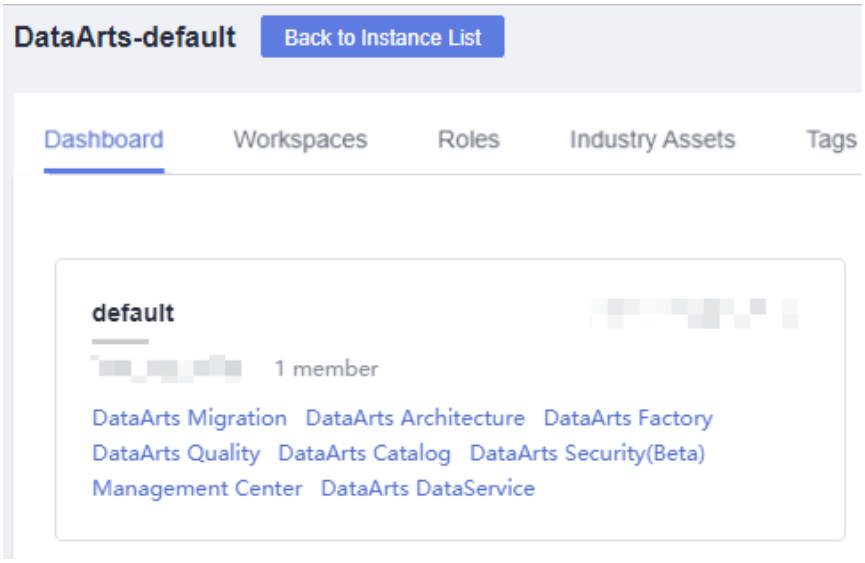
Figure 9-227 Data security diagnosis



You are advised to scan data at least once a month to ensure data security and reliability. The procedure of diagnosing data security risks is as follows:

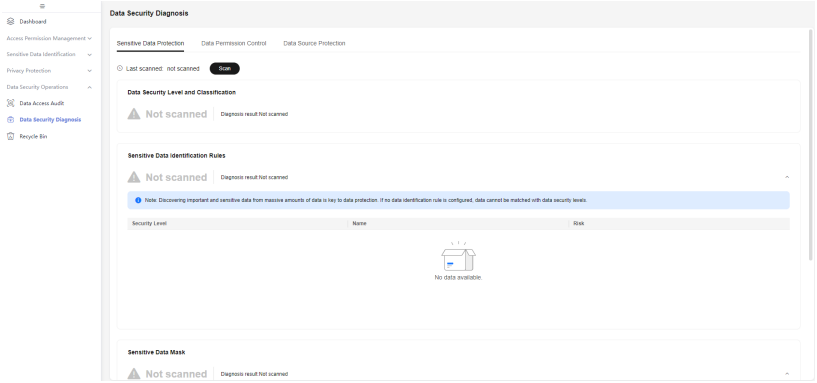
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-228 DataArts Security



Step 2 In the navigation pane on the left, choose **Data Security Diagnosis**.

Figure 9-229 Data Security Diagnosis

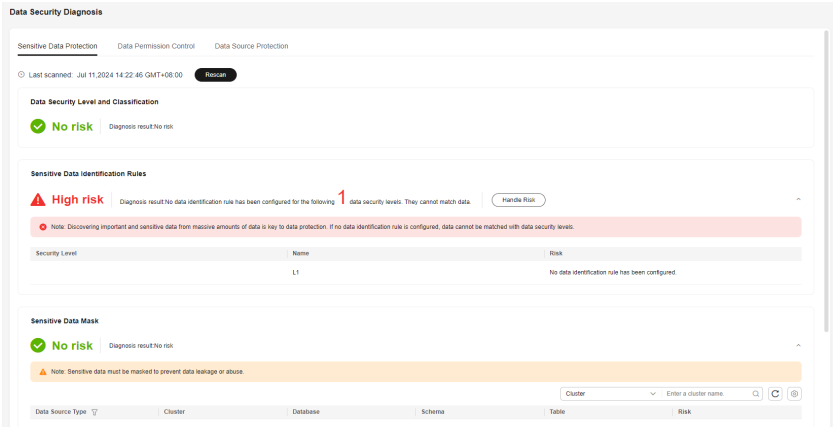


Step 3 Click the **Sensitive Data Protection**, **Data Permission Control**, or **Data Source Protection** tab, and click **Scan** or **Rescan**.

Step 4 After the scan is complete, identify risky items based on the scan result and handling suggestions and click **Handle Risk** to ensure data security and reliability.

You are advised to handle medium and high security risks as soon as possible. The following figure shows the risk level and diagnosis result of a check item on the **Sensitive Data Protection** page.

Figure 9-230 Security diagnosis result



----End

9.7 Managing the Recycle Bin

The recycle bin allows you to restore key data of DataArts Security that has been deleted by mistake. The key data includes permission set–related resources (workspace permission sets, permission sets, and common roles) and dynamic masking policies. The key data is determined by the importance, use frequency, and restoration difficulty of data.

Prerequisites

Permission set–related resources (workspace permission sets, permission sets, and common roles) or dynamic masking policies have been deleted in the last 30 days.

Notes and Constraints

- Only the DAYU Administrator, Tenant Administrator, and data security administrator can restore data.
- Managed MRS roles are existing roles in MRS data and are not defined in DataArts Security, so they will not be moved to the recycle bin when deleted.
- After permission set–related resources and dynamic masking policies are deleted and moved to the recycle bin, their synchronization statuses will become unsynchronized. After they are restored from the recycle bin, they must be synchronized so that they can take effect.
- Data in the recycle bin can be retained for a maximum of 30 days. Deleted data will be permanently cleared after a 30-day retention period.
- A maximum of 1,000 permission sets or dynamic masking policies can be retained in the recycle bin of an instance. If that limit is exceeded, the oldest permission sets or dynamic masking policies will be automatically cleared first, on a first-in-first-out basis.
- If **Name Conflict Strategy** is set to **Add a timestamp to each name** during data restoration and the name of the data to be restored already exists, a timestamp will be added to the name of the data to be restored. That is, the name of the restored data is in ***Original name_13-digit timestamp*** format. If

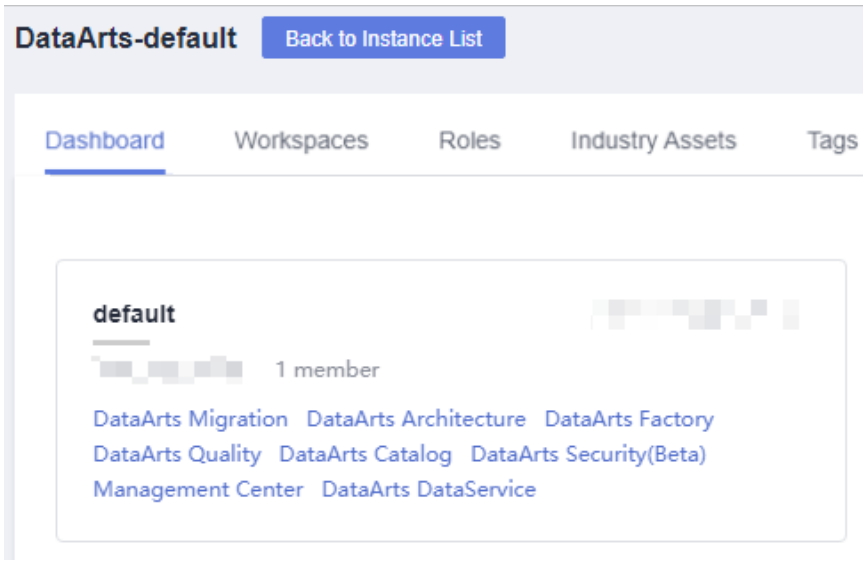
the name of the data to be restored with the timestamp contains more than 64 characters, the original name will be truncated to ensure that the name of the data to be restored contains no more than 64 characters.

- When you restore a permission set that was deleted by mistake from the recycle bin, the association between permission sets will be checked. If certain conditions are not met, the permission set cannot be restored. For example, if the parent permission set of a permission set has been deleted, the permission set can be restored only after its parent permission set is restored.
- A maximum of 20 data records can be restored at a time.

Restoring Data in the Recycle Bin

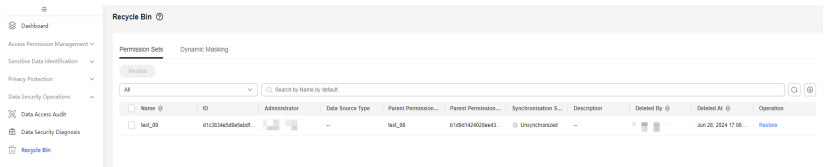
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Security**.

Figure 9-231 DataArts Security



Step 2 In the left navigation pane, choose **Recycle Bin**.

Figure 9-232 Recycle Bin page

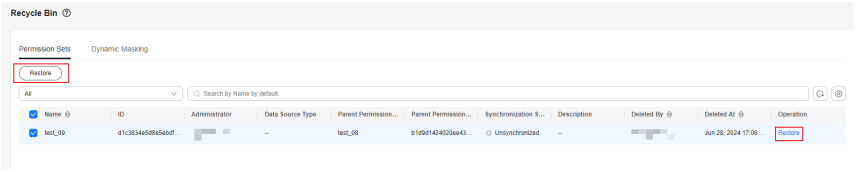


Step 3 On the **Recycle Bin** page, you can view and restore deleted permission set–related resources (workspace permission sets, permission sets, and common roles) and dynamic masking policies.

The operations for restoring different types of data are similar. In the following operations, permission sets are used as an example to describe how to restore data.

Step 4 On the **Permission Sets** page, locate the permission set you want to restore and click **Restore** in the **Operation** column. Alternatively, select the permission sets you want to restore and click **Restore** above the list to restore the permission sets.

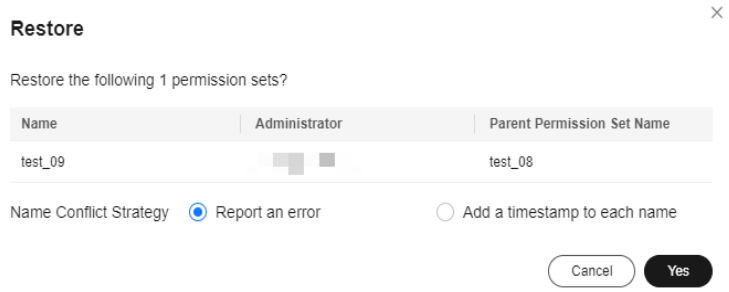
Figure 9-233 Restoring data



Step 5 In the displayed dialog box, set **Name Conflict Strategy** to avoid a conflict between the restored data and existing data. Then click **Yes**.

- **Report an error**: If the name of the data to be restored already exists, an error will be reported and the data will not be restored.
- **Add a timestamp to each name**: If the name of the data to be restored already exists, a timestamp will be added to the name. That is, the name of the data to be restored is in ***Original name_13-digit timestamp*** format. If the name of the data to be restored with the timestamp contains more than 64 characters, the original name will be truncated to ensure that the name of the data to be restored contains no more than 64 characters.

Figure 9-234 Setting Name Conflict Strategy



Step 6 After restoring workspace permission sets, permission sets, common roles, or dynamic masking policies, check them on corresponding pages and synchronize them to make them take effect.

----End

10 DataArts DataService

10.1 Overview

DataArts Studio DataArts DataService aims to build a unified data service bus for enterprises to centrally manage internal and external API services. DataArts DataService helps you quickly generate data APIs based on data tables and allows you manage the full lifecycle of APIs, covering API publishing, management, and O&M. With DataArts DataService, you can implement microservice aggregation, frontend-backend separation, and system integration, and provide functions and data for partners and developers easily and quickly at a low cost and risk.

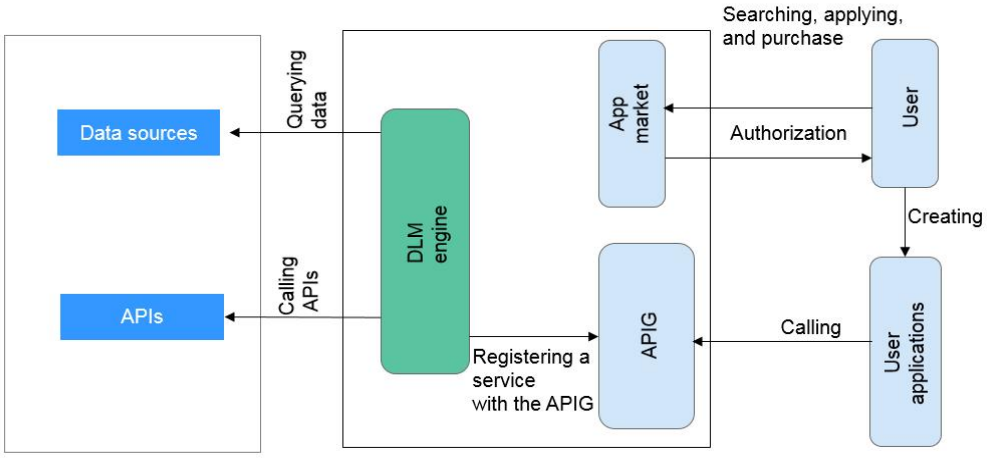
DataArts DataService has the following advantages over other data sharing and exchange methods:

- Unified interface standards reduce the workload for interconnection with upper-layer applications.
- Data logic is deployed on the data platform and is therefore decoupled from the application logic. This reduces repeated development of data models and avoids frequent changes caused by data logic adjustment.
- Data logic-related storage and compute resources are deployed on the data platform, reducing resource consumption on applications.
- A large amount of detailed and sensitive data is inaccessible to applications. In addition, DataArts DataService improves data security by means of API review and publishing, authentication and throttling, and dynamic anonymization.

DataArts DataService encapsulates data logic into RESTful APIs of a unified standard that can be used to access data. DataArts DataService applies to quick response to the requests for accessing a small amount of data. To open a large amount of data, you are advised to adopt data sharing and exchange or other solutions. .

DataArts DataService uses the serverless architecture. You only need to focus on the API query logic and do not need to worry about the infrastructure such as the operating environment. DataArts DataService prepares compute resources, supports elastic scaling, and spares O&M expenditure.

Figure 10-1 DataArts DataService architecture



Publishing an API

To publish an API or a group of APIs, do as follows:

1. **Make preparations**.

   If you want to use DataArts DataService, you must perform the operations in **Buying an Exclusive DataArts DataService Instance**.

   In addition, before creating an API, you must add a reviewer by following the instructions in **Adding Reviewers**.

2. **Create an API**.

   You can **generate** and **register** APIs. An API can be generated in the **wizard mode** or **script mode**.

3. **Debug the API**.

   Debug the created API on the management console to check whether it runs properly.

4. **Publish the API**.

   The API can be called only after it is published.

5. **(Optional) Manage the API**.

   You can manage the published API as needed.

6. **(Optional) Perform throttling**.

   To ensure the stability of backend services, you can perform throttling on the API.

Calling an API

To call an API, perform the following operations:

1. Obtain an API.

   Obtain the API from the service catalog. An API can be called only after it is published.

2. (Optional) Create an application and get authorized.

For an API that is accessed using application authentication, you need to **create an application** and **authorize the application to use the API**. When you call an API, DataArts DataService verifies your identity based on the key pair (AppKey and AppSecret) of the created application.

3. **Call the API**.

   After completing the preceding steps, you can call the API.

Overview Page

On the **Overview** page, you can view various monitoring data views. The **Overview** page displays **Develop APIs** and **Call APIs**.

Figure 10-2 Develop APIs tab page

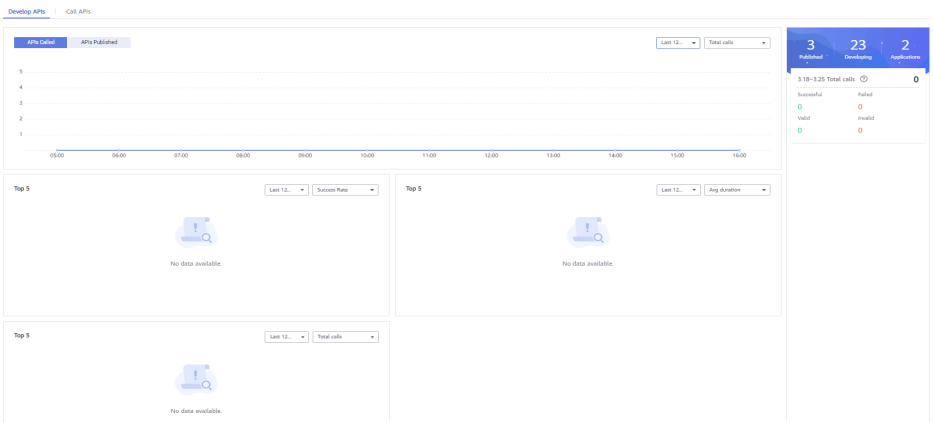


Table 10-1 Parameters on the Develop APIs tab page

Parameter	Description
APIs Published	The number of APIs published every day, week, month, and year.
APIs Called	The number of times that APIs are called in half a day, every day, every week, and every month.
Top 5 (1)	The call rate of APIs, including the success rate, failure rate, validity rate, and invalidity rate.
Top 5 (2)	The calling duration of APIs, average duration, success duration, and failure duration.
Top 5 (3)	The top 5 APIs that are called, successful API calls, failed API calls, valid API calls, and invalid API calls.
Published	The number of APIs that have been published.
Developing	The number of APIs that are being developed.
Applications	The number of APIs that are requested by applications.
Successful	The number of successful API calls.
Failed	The number of failed API calls.

Parameter	Description
Total	The total number of API calls.

Figure 10-3 Call APIs tab page

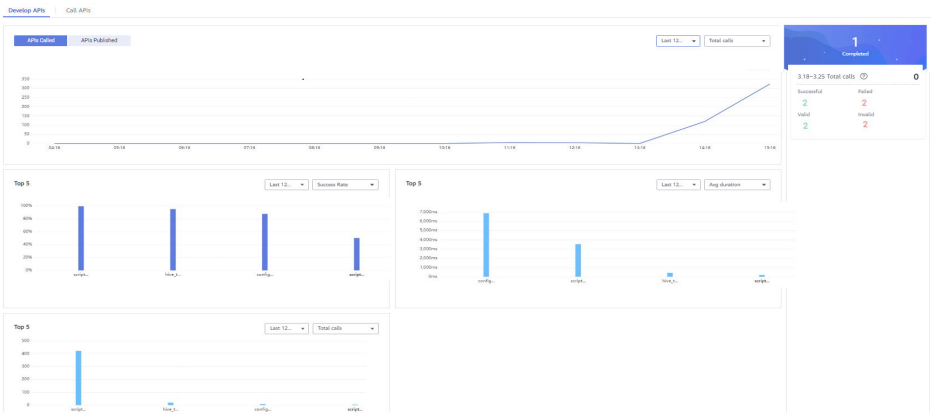


Table 10-2 Parameters on the Call APIs tab page

Parameter	Description
APIs Called	The number of API calls made every day, week, month, and year.
Top 5	The ratio of successful and failed API calls in the last seven days.
Completed	The number of APIs applied on the DataArts DataService platform.
Successful	The number of successful API calls on the DataArts DataService platform.
Total	The number of total API calls on the DataArts DataService platform.

10.2 Specifications

Specifications of Exclusive DataArts DataService

Table 10-3 lists the specifications of DataArts DataService Exclusive.

Table 10-3 Specifications of Exclusive DataArts DataService

Instance	Max. APIs That Can Be Published	Delay (Unit: ms)
Small	500	<20
Medium	1,000	<15

Instance	Max. APIs That Can Be Published	Delay (Unit: ms)
Large	2,000	<10

Specifications of API Return Data

DataArts DataService is applicable to interactions involving a small amount of data, and is not applicable to returning a large amount of data through APIs. The following table lists the specifications of the data returned by DataArts DataService APIs.

Table 10-4 Restrictions on the number of data records returned by an API

API Category	Scenario	Data Source	Default Specifications
Configuration	Debugging	DLI/ MySQL/RDS/DWS	10
	Call	DLI/ MySQL/RDS/DWS	100
Script	Test SQL	N/A	10
	Debugging	DLI	• Default pages: 100 • Custom pages: 1,000
		MySQL/RDS/DWS	• Default pages: 10 • Custom pages: 2,000
	Call	DLI	• Default pages: 100 • Custom pages: 1,000
		MySQL/RDS/DWS	• Default pages: 10 • Custom pages: 2,000

10.3 API Development

10.3.1 Preparations

10.3.1.1 Buying an Exclusive DataArts DataService Instance

This topic describes how to buy an exclusive DataArts DataService instance. You can create an API in Exclusive DataArts DataService and use it to provide services only after the instance is available.

NOTICE

To create or delete an exclusive cluster or change API quotas, you must have either of the following accounts:

- Administrator with the VPCEndpoint Administrator permission
- Tenant Administrator with the VPCEndpoint Administrator permission

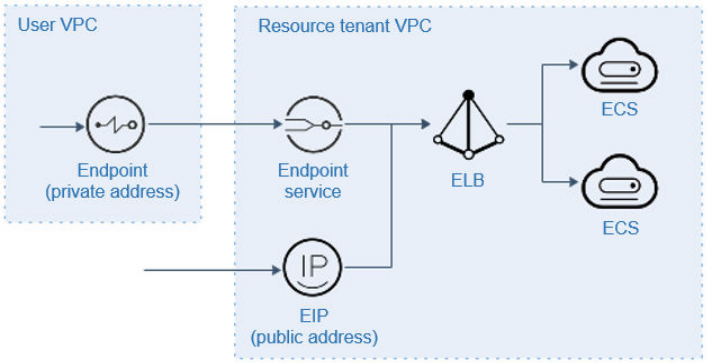
Network Environment Preparation

After a DataArts DataService exclusive cluster is created, resources are located in the resource tenant zone. ELB performs load balancing for the nodes in the cluster.

You can access the cluster in either of the following ways:

- Private address: IP address of the VPC endpoint
- Public address (optional): EIP bound to ELB The EIP is available only when you enable the Internet access when creating the DataArts DataService cluster.

Figure 10-4 Networking of the DataArts DataService exclusive cluster



To ensure that the created exclusive cluster is accessible, pay attention to the following network configurations:

- Virtual Private Cloud (VPC)

  A VPC must be configured for an exclusive DataArts DataService instance. Resources (such as ECSs) in the same VPC can use the private address of the exclusive instance to call APIs.

  When you buy an exclusive instance, you are advised to configure the same VPC as other associated services to ensure network security and facilitate network configuration.
- Elastic IP (EIP)

  If you want to call an API of an exclusive instance, buy an EIP and bind it to the instance. The EIP will be used as the Internet entry of the instance.
- Security Group

  A security group is similar to a firewall. It controls who can access the specified port of an instance and enables the communication data flow of the instance to move to the specified destination address. You are advised to

enable the IP address and port in the inbound direction of the security group to protect the network security of the instance to the maximum extent.

The security group bound to an exclusive instance must meet the following requirements:

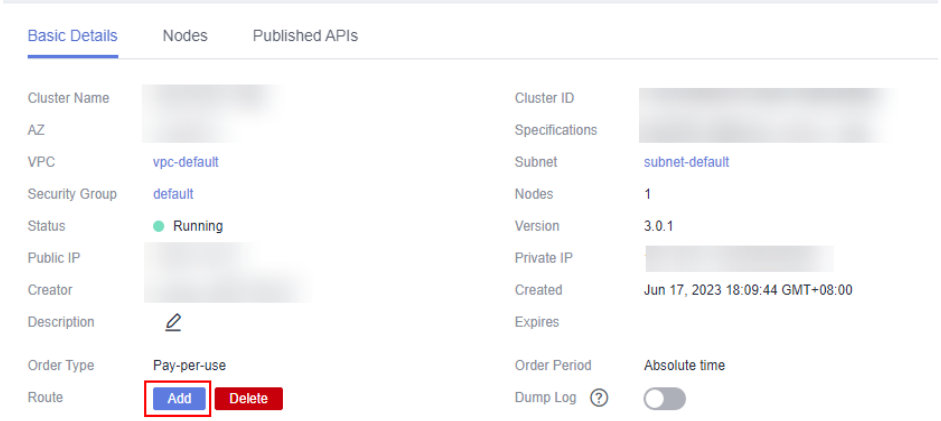
- Inbound rule: To call APIs from the Internet or from resources in other security groups, enable ports 80 (HTTP) and 443 (HTTPS) in the inbound direction of the security group bound to the exclusive instance.
- Outbound direction: If the backend service is deployed on the Internet or in another security group, enable the backend service address and API calling listening port in the outbound direction of the security group bound to the exclusive instance.
- If the frontend and backend services of the API are bound to the same security group and VPC as the exclusive instance, you do not need to enable the preceding ports for the exclusive instance.

- Route

  In the physical machine management scenario, if the physical machine and the cluster have different network segments, you need to configure a route.

  On the **Basic Details** page, click **Add** following **Route** and add the IP address of the physical server.

Figure 10-5 Basic Details page



Procedure

After you buy a DataArts DataService incremental package, the system automatically creates a cluster based on your selected specifications.

Step 1 Locate an enabled instance and click **buy**.

Step 2 On the displayed page, set parameters based on **Table 10-5**.

Table 10-5 Parameters for buying an exclusive DataArts DataService instance

Parameter	Description
Package	Select **DataArts DataService**.

Parameter	Description
Billing Mode	Currently, **Yearly/Monthly** is supported.
Workspace	The workspace for which you want to use the incremental package. For example, if you want to use DataArts DataService Exclusive in workspace A of the DataArts Studio instance, select workspace A. After you buy an exclusive DataArts DataService cluster, you can view it in workspace A.
AZ	When you buy a DataArts Studio instance or incremental package for the first time, you can select any available AZ. When you buy another DataArts Studio instance or incremental package, determine whether to deploy your resources in the same AZ based on your DR and network latency demands. • If your application requires good DR capability, deploy resources in different AZs in the same region. • If your application requires a low network latency between instances, deploy resources in the same AZ. For details, see **AZs**.
Name	N/A
Description	A description of the exclusive DataArts DataService cluster.
Version	Cluster version of the exclusive DataArts DataService cluster.
Cluster Details	The number of concurrent API requests supported varies depending on the instance specifications.
Enabling public IP address	If you select **Enabling public IP address**, external services can call the APIs created in exclusive instances through the Internet address.
Bandwidth	Bandwidth range on the Internet.
VPC	A VPC is a secure, isolated, and logical network environment. Cloud resources (such as ECSs) within the same VPC can call APIs using the private IP address of DataArts DataService Exclusive. Deploy the DataArts DataService Exclusive instance in the same VPC as your other services to facilitate network configuration and secure network access. **NOTE** After the DataArts DataService instance is created, the VPC cannot be changed.

Parameter	Description
Subnet	A subnet provides dedicated network resources that are logically isolated from other networks for network security. Deploy the DataArts DataService Exclusive instance in the same subnet of the same VPC as your other services to facilitate network configuration and secure network access. **NOTE** After the DataArts DataService instance is created, the subnet cannot be changed.
Security Group	A security group is used to set port access rules, define ports that can be accessed by external services, and determine the IP addresses and ports that can be accessed externally. For example, if the backend service is deployed on an external network, configure security group rules to allow access to the IP address and listening port of the backend service. **NOTE** 1. If **Enabling the public IP address** is selected, the security group must allow access from ports 80 (HTTP) and 443 (HTTPS) in the inbound direction. 2. After the DataArts DataService instance is created, the security group cannot be changed.
Managing Cluster Resources Using an Enterprise Project	Enterprise project associated with the exclusive DataArts DataService cluster. An enterprise project facilitates management of cloud resources. For details, see **Enterprise Management User Guide**.
Nodes	N/A
Required Duration	N/A

Step 3 Click **buy Now**, confirm the settings, and click **Next**.

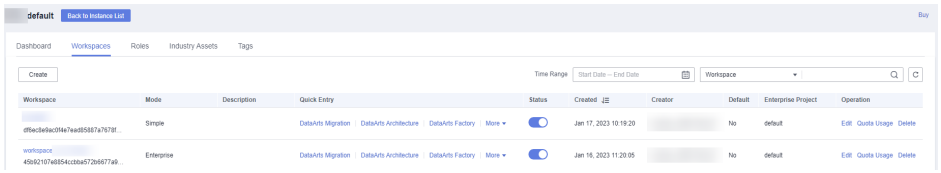
----End

Setting the Allocated API Quota

After creating an exclusive cluster, you need to set the allocated API quota so that you can create APIs. To set the quota, perform the following steps:

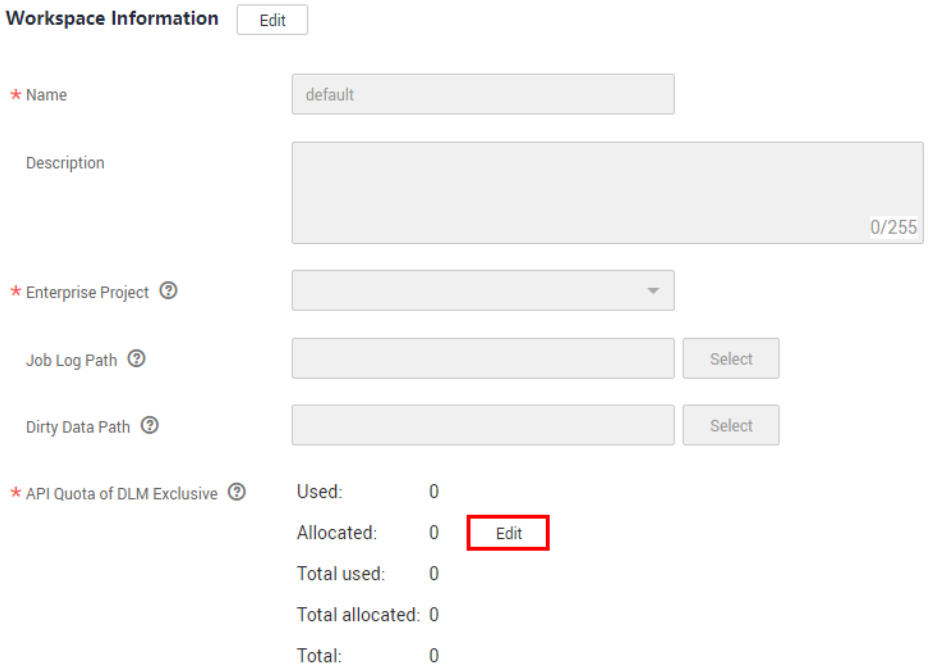
Step 1 On the **Workspaces** page, locate a workspace and click **Edit** in the **Operation** column.

Figure 10-6 Editing a workspace



Step 2 In the displayed **Workspace Information** dialog box, click **Edit** to set the allocated quota.

Figure 10-7 Setting the allocated quota

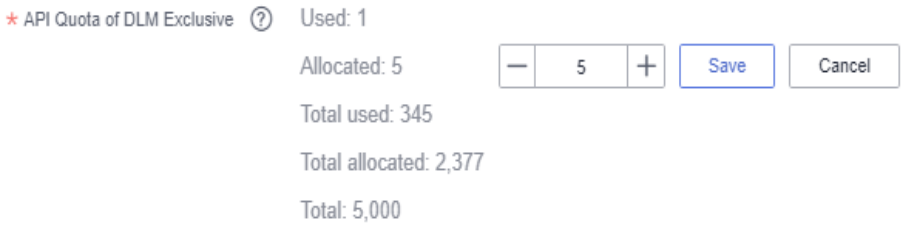


NOTE

You can create 10 DataArts DataService Exclusive APIs for free in each DataArts Studio instance, and you will be charged for each extra API.

Step 3 Set the allocated API quota for DataArts DataService Exclusive.

Figure 10-8 Setting the quota



NOTE

The allocated quota cannot be less than the used quota and not greater than the total quota minus the total allocated quota plus the previously allocated quota.

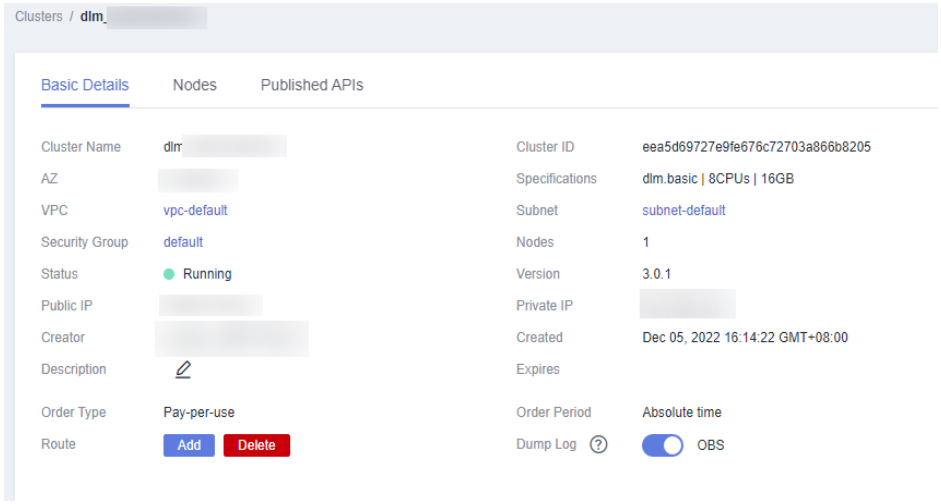
----End

Setting Log Dump

After obtaining a cluster, you can set log dump. When this function is enabled, all the API access logs in the current workspace of the cluster will be dumped to a specified OBS bucket or LTS log.

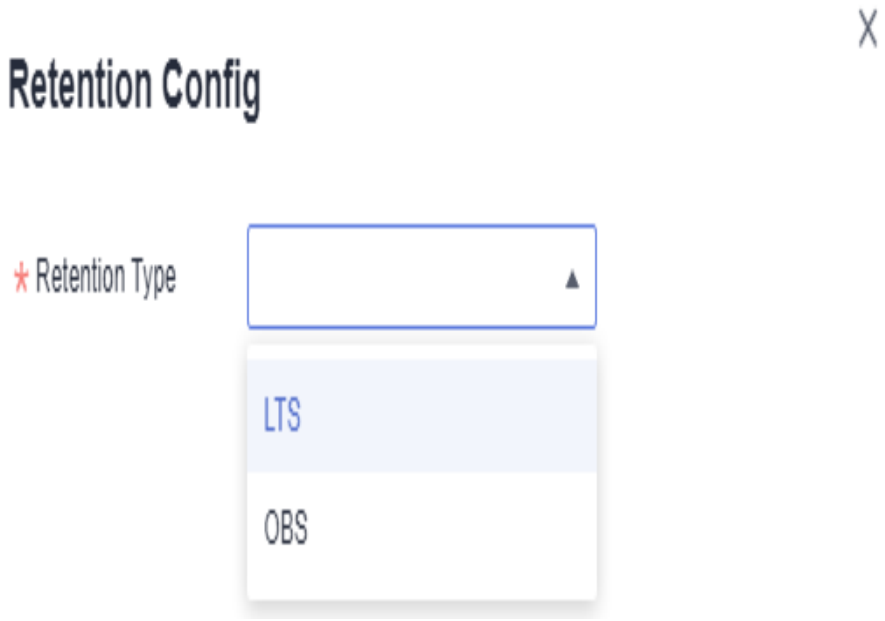
Step 1 On the page displaying clusters, click a cluster name to go to the **Basic Details** tab page.

Figure 10-9 Basic Details



Step 2 Enable **Dump Log** and select a retention type (**OBS** or **LTS**).

Figure 10-10 Selecting a retention type



Step 3 If you select **OBS**, all the API access logs in the current workspace will be dumped to the specified OBS bucket.

Step 4 If you want to select **LTS**, you need to create a log group and a log stream on the LTS console in advance. For details about how to create a log group and a log stream, see **Configuring Log Dump and Viewing Logs on LTS**. When you select **LTS**, all the API access logs in the current workspace will be dumped to the created log stream.

----End

10.3.1.2 Adding Reviewers

APIs must be reviewed and approved before they can be published. APIs are reviewed in the following way:

- An API publisher who does not have the reviewer permission must submit the API to the reviewer for review.
- An API publisher who has the reviewer permission can publish an API without review or approval.

Therefore, if you do not have the reviewer permission and want to publish an API, you must add a reviewer first. Only the workspace admin has the permissions required to add reviewers.

NOTE

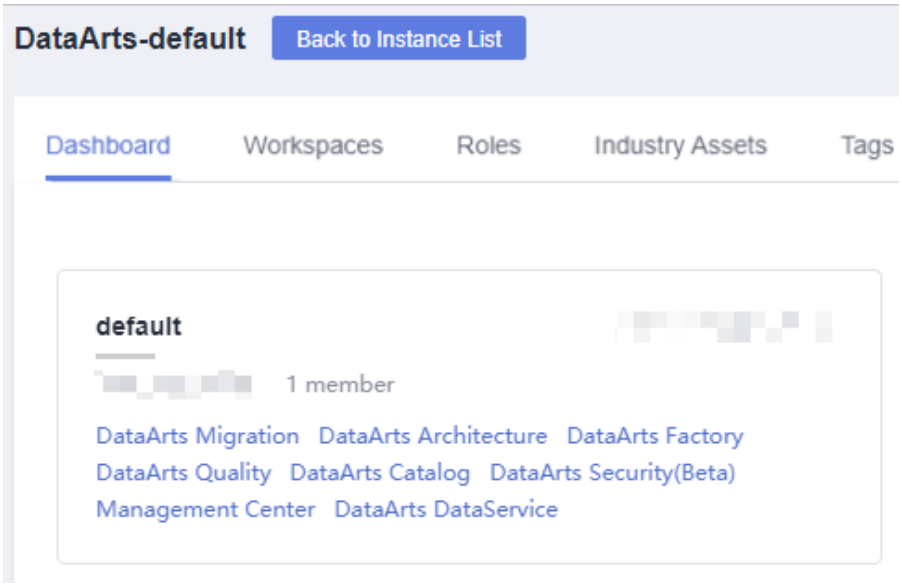
An admin, developer, or operator can be a reviewer. A viewer cannot be a reviewer.

Regardless of whether they are added as reviewers, users with the admin role of the workspace have the reviewer permissions by default.

Procedure

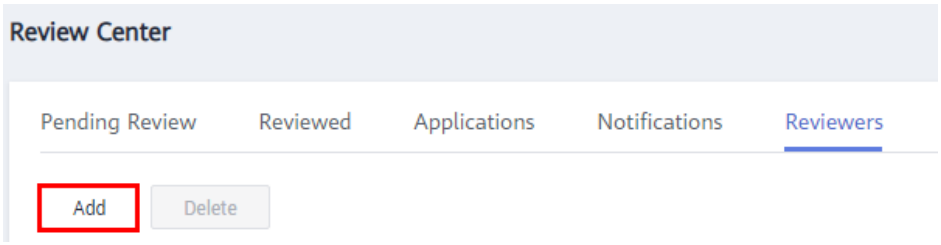
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

   Figure 10-11 DataArts DataService

2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **Operation Management** > **Review Center** from the left navigation pane. On the page displayed, choose **Reviewer Management** and click **Add**.

   Figure 10-12 Adding reviewers

4. Select a reviewer (workspace member), enter a correct phone number and email address, and click **OK**.
5. Add more reviewers, if required.

10.3.2 Creating an API

10.3.2.1 Generating an API Using Configuration

This section describes how to generate an API using configuration.

Generating data APIs using configuration is simple. You do not need to write any code. Wizard mode is designed for users who do not have high requirements on API functions or have no experience in code development.

Prerequisites

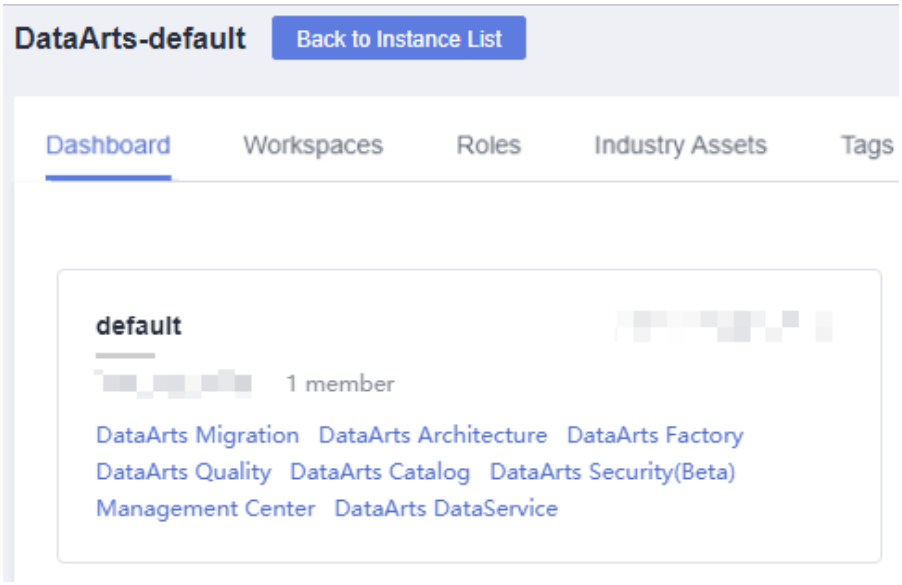
You have configured data sources on the **Data Connection Management** page of **Management Center**.

Creating an API Directory

An API catalog is an API index that is orchestrated and recorded in a certain sequence. It is a tool for reflecting categories, guiding API usage, and searching for APIs, helping API developers effectively classify and manage API services.

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

   Figure 10-13 DataArts DataService

2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **API Catalogs** and click ⊕.

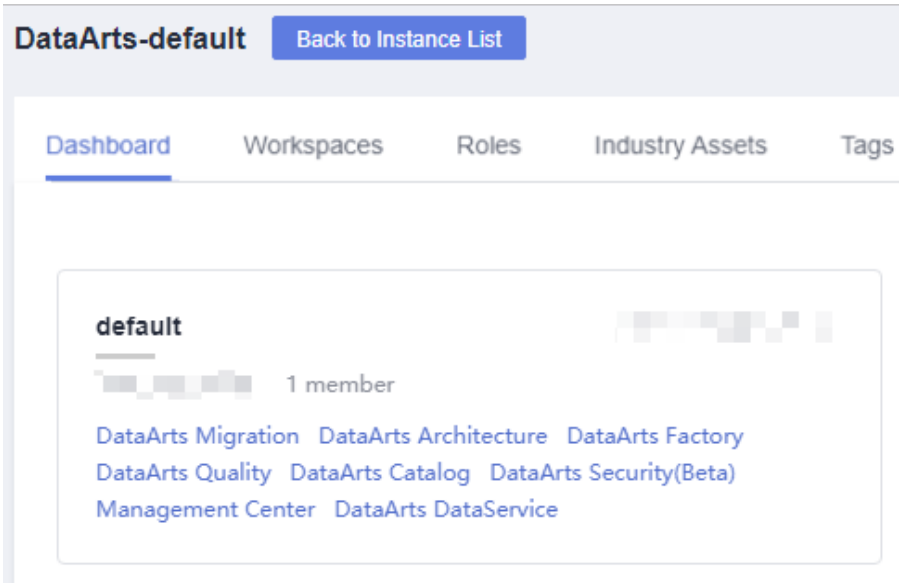
   In the dialog box displayed, enter an API catalog name, and click **OK**.
4. In the **Operation** column of an API catalog, edit or manage the API catalog.

   Click **Edit** to the right of the API catalog that you want to edit. An API can be edited only when it is in the **Created**, **Rejected**, **Offline**, or **Disabled** state.

Configuring Basic API Information

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-14 DataArts DataService



2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **APIs** from the left navigation bar, and click **Create**. On the displayed page, enter the basic information.

Table 10-6 API basic configuration

Parameter	Description
API	An API name consists of 3 to 64 characters and starts with a letter. Only letters, numbers, and underscores (_) are allowed.
API Catalog	A collection of APIs for a specific function or scenario. It is similar to a folder and specifies the location of APIs. You can search for APIs in a specified API catalog. The API catalog is the minimum organization unit of APIs in DataArts DataService and also the minimum management unit in the API gateway. Click **Select Catalog** to create an API catalog or select an existing one created in **Creating an API Directory**.

Parameter	Description
Request Path	API access path, for example, **/getUserInfo** It is the part between the domain name and query parameters in the URL of a request path, for example, **/blogs/*xxxx*** shown in the following figure. **Figure 10-15** API access path in the URL  Braces ({}) can be used to identify parameters in a request path as wildcard characters. For example, **/blogs/{blog_id}** indicates that any parameter can follow **/blogs**. **/blogs/188138** and **/blogs/0** can both match **/blogs/{blog_id}**, and are processed by this API. In addition, duplicate request paths are not allowed for the same domain name. When a path parameter is used as a wildcard, the name is not unique. For example, **/blogs/{blog_id}** and **/blogs/{xxxx}** are considered as the same path.
Parameter Protocol	Protocol used to transmit requests. The shared edition supports HTTP and HTTPS, and the exclusive edition supports HTTPS. HTTPS is recommended. HTTP is insecure and may have security risks. • HTTP is a basic network transmission protocol. It is stateless, connectionless, simple, fast, and flexible, and uses plaintext for transmission. It is easy to use but has poor security. • HTTPS is an HTTP-based protocol with SSL or TLS encryption verification. It can effectively verify identities and protect data integrity. To access HTTPS APIs, you need to configure related SSL certificates or skip SSL verification.
Request Method	HTTP request method, indicating the type of the requested operation, such as GET and POST. The method complies with the resultful style. • **GET** requests the server to return specified resources. This method is recommended. • **POST** requests the server to add resources or perform special operations. This method is used only for API registration. The POST request does not have a body. Instead, it involves transparent transmission.
Description	A brief description of the API to create.

Parameter	Description
Tag	API tag. The tag is used to mark the API attributes. After the API is created, you can quickly search for the API by tag. A maximum of 20 tags can be set for an API.
Reviewer	A reviewer who has permissions to review APIs. Click **Add** to enter the **Review Center** page and click **Add** on the **Reviewers** tab page to add a reviewer.
Security Authentication	When creating an API, you can select one of the following authentication modes. The three modes differ in how the API is called. You are advised to use **App Authentication**, which is more secure that the other two modes. • **App authentication**: App authentication is used for calling an API. The AppKey & AppSecret is used for authentication. It is highly secure. When **App authentication** is used, an SDK is required for access. Java, Go, Python, JavaScript, C#, PHP, C++, C, and Android SDKs are available. For details about how to call APIs in each language, see **Calling APIs Through App Authentication**. • **IAM authentication**: IAM authenticates API requests. This mode is available only for Huawei cloud users. The security level is medium. When using IAM authentication, you need to call the **Obtaining a User Token** API of IAM to obtain a token, add the **X-Auth-Token** parameter with the obtained token as the value to the request header, and use an API calling tool or SDK to call released APIs. • **Non-authentication**: No authentication is required. This mode allows all users to access APIs, which may pose security risks. It is recommended only for testing APIs. If the caller is not a trusted user, there is a risk of data leakage, breakdowns caused by high concurrent access, SQL injection, and others. This mode does not require any authentication information. You can use an API calling tool or SDK to directly call an API by specifying required parameters.
Display Scope	After the API is published, all users in the selected scope can view the API in the service catalog. • Current workspace APIs • Current project APIs • Current tenant's APIs
Access Log	If you select this option, the API query result will be recorded and retained for seven days. You can choose **Operations Management** > **Access Logs** and select the request date to view the logs.

Parameter	Description
Min. Retention Period	Minimum retention period of the API publishing status, in hours. Value **0** indicates that the retention period is not limited. You can suspend, unpublish, or cancel authorization for an API only after the minimum retention period ends. The system notifies the authorized users. If all authorized users have processed the notifications or unbound the API from their apps, the API will be suspended or unpublished, or the API authorization will be canceled. Otherwise, the system will forcibly suspend, unpublish, or cancel authorization for the API when the minimum retention periods ends. For example, if the minimum retention period is set to 24 hours, the API can be suspended 24 hours after it is published. If the authorized user handles the notifications in the review center or unbind the API from the app, the API will be directly suspended. Otherwise, the API will be forcibly suspended when the minimum retention period ends.

Parameter	Description
Input Parameters	Parameters required for calling the API. The parameters are used as the request parameters on the **Set Data Extract Logic** page. An input parameter consists of the parameter location, parameter type, whether the parameter is mandatory, and the default value. ● The parameter location can be **Query**, **Header**, **Path**, or **Body**. In addition, static parameters are supported. – **Query** is the query parameter following the URL. It starts with a question mark (?) and connects multiple parameters with **&**. – **Header** is located in the request header and is used to transfer current information, for example, host and token. – **Path** is a request parameter in the request path. If you configure a path parameter, you must also add this parameter to the request path. – **Body** is a parameter in the request body and is generally in JSON format. – **Static** is a static parameter that does not change with the value passed by API callers. The parameter value is determined upon API authorization. If the parameter value is not set during authorization, the default value of the API input parameter is used. ● The parameter type can be **Number** or **String**. **Number** corresponds to numeric data types such as int, double, and long. **String** corresponds to text data types such as char, vachar, and text. ● **Mandatory** and **Default Value**: If you select **Yes** for **Mandatory**, parameters must be passed for accessing the API. Otherwise, the default value of the parameter will be used if the parameter is not passed for accessing the API. **NOTE** When defining an input parameter, ensure that the following size requirements are met: ● **Query** and **Path**: 32 KB. ● **HEADER**: The maximum size is 128 KB. ● **BODY**: The maximum size is 128 KB. You need to set input parameters based on the designed request parameters for the API. For example, the request path of the API used to query user information in a table by user ID is **/getUserInfo**. You can configure input parameters as follows: ● If the request parameter for calling the API is **id**, and the information about the user with **id** needs to be returned, configure an input parameter as follows:

Parameter	Description
	1. Click **Add** and enter **id** for **Name**. 2. Set **Parameter Location** to **Query**. 3. Set **Type** to **Number**. 4. Select **Yes** for **Mandatory**. 5. Retain the default value. • If the request parameters for calling the API are **id1** and **id2**, and the user information between **id1** and **id2** needs to be returned, configure input parameters as follows: 1. Click **Add** and enter **id1** for **Name**. 2. Set **Parameter Location** to **Query**. 3. Set **Type** to **Number**. 4. Select **Yes** for **Mandatory**. 5. Retain the default value. 6. Click **Add** again and configure parameter **id2**.

4. After the basic API information is complete, click **Next** to go to the **Data Extract Logic** page.

Configuring the Data Extraction Logic

Set **Data Acquisition Method** to **GUI based**.

1. Select a data source, data connection, database, and data table to obtain the tables to be configured.

   NOTE

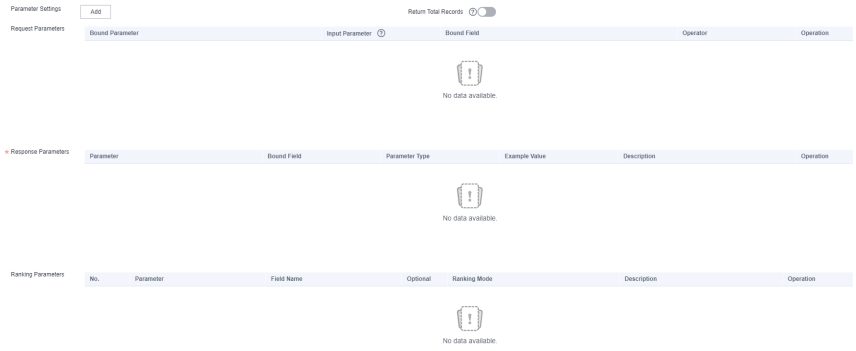
   For details on the data sources supported by DataArts DataService, see **Data Sources**. Configure data sources in Management Center in advance. You can search for a data table by name.

2. Configure parameter fields.

   Click **Add** next to **Parameter Settings**. All fields in the table are displayed on the page for adding parameters. Select the request parameters, response parameters, and ranking parameters that you want to add to the corresponding lists.

   In addition, you can enable **Return Total Records**. Then the total number of script execution results will be returned.

Figure 10-16 Add Parameter dialog box



3. Edit request parameters.

   A request parameter consists of a bound parameter, bound field, and operator. In the request parameter list, select a bound parameter and an operator.

   - Bound parameters are available to external systems. They are the input parameters defined on the **Configure Basic Details** page and are directly used to access the API.
   - Bound fields are invisible to external systems. They are fields of the selected tables and are accessed during an API call.
   - Operators determine how bound fields and parameters in access requests are processed. A bound field is on the left of an operator and a bound parameter is on the right. The following table lists the available operators.

     Table 10-7 Available operators

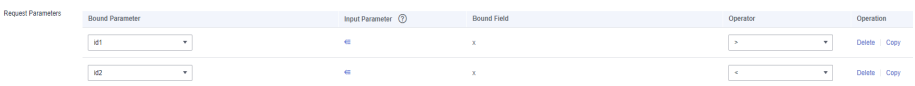
Operator	Description
=	Checks whether the values of two operands are equal. The condition is true if the bound field is equal to the bound parameter.
<>	Checks whether the values of two operands are equal. The condition is true if the bound field is not equal to the bound parameter.
>	Checks whether the value of the left operand is greater than that of the right operand. The condition is true if the bound field is greater than the bound parameter.
>=	Checks whether the value of the left operand is greater than or equal to that of the right operand. The condition is true if the bound field is greater than or equal to the bound parameter.

Operator	Description
<	Checks whether the value of the left operand is less than that of the right operand. The condition is true if the bound field is less than the bound parameter.
<=	Checks whether the value of the left operand is less than or equal to that of the right operand. The condition is true if the bound field is less than or equal to the bound parameter.
%like%	Ignores the prefix and suffix in character matching. The condition is true if the bound field (excluding the prefix and suffix) can match the bound parameter.
%like	Ignores the prefix in character matching. The condition is true if the bound field (excluding the prefix) can match the bound parameter.
like%	Ignores the suffix in character matching. The condition is true if the bound field (excluding the suffix) can match the bound parameter.
in	Compares a value with a specified list of values. The condition is true if the bound field can match the values in multiple bound parameters.
not in	Compares a value with values not in a specified list. It is the opposite of the in operator. The condition is true if the bound field cannot match the values in multiple bound parameters.

You can copy and set operators in request parameters to match input bound parameters with bound fields.

As shown in **Figure 10-17**, you can enter parameters **id1** and **id2** when accessing an API to match the values of **x** columns between **id1** and **id2**.

Figure 10-17 Request parameters



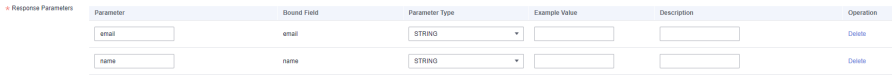
4. Edit response parameters.

   A response parameter consists of the parameter name, bound field, and parameter type.

   – Parameters are available to external systems and can be customized. They are returned to API callers.

- Bound fields are invisible to external systems. They are fields of the selected tables and are returned during an API call.
- The parameter type is the data display format when the API is called, and can be a numeric or character.

Figure 10-18 Response parameters

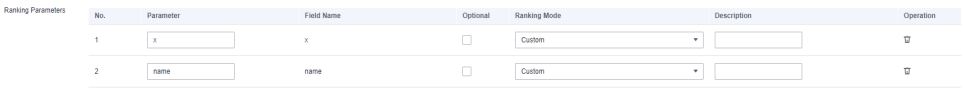


5. Edit ranking parameters.

   A ranking parameter consists of the parameter name, field name, whether the parameter is optional, and ranking mode. Multiple ranking parameters are allowed. If there are multiple ranking parameters and the first ranking parameter is the same, the subsequent ranking parameters are used one by one. You can click **Add** next to **Parameter Settings** and adjust the sequence of ranking parameters.

   - Parameter names can be customized and associated with field names.
   - Field names are invisible to external systems. They are fields of the selected tables and are accessed during an API call.
   - If you select **Optional**, the parameter is optional.
   - The ranking mode can be ascending, descending, or custom. If you set **Ranking Mode** to **Ascending** or **Descending**, but set **pre_order_by** to a value different from the value of **Ranking Mode** when testing or calling the API, the API cannot be called.

   Figure 10-19 Ranking parameters



6. Click **Next** to go to the API test page.

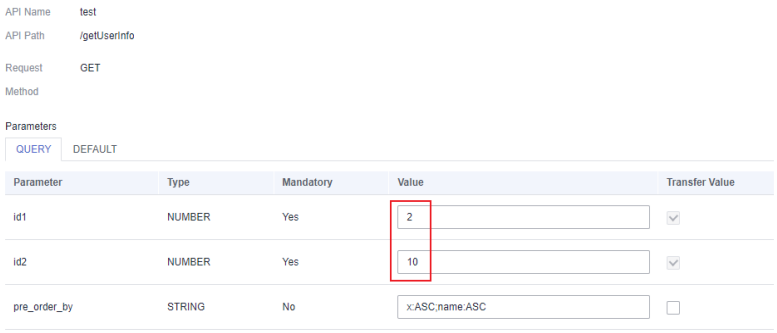
Testing the API

1. Set values for input parameters.

   If you want to set multiple values for a parameter, observe the following format:

   - String: 'a','b','c'
   - Value: 1,2
   - Field: a,b,c

Figure 10-20 Setting values for input parameters



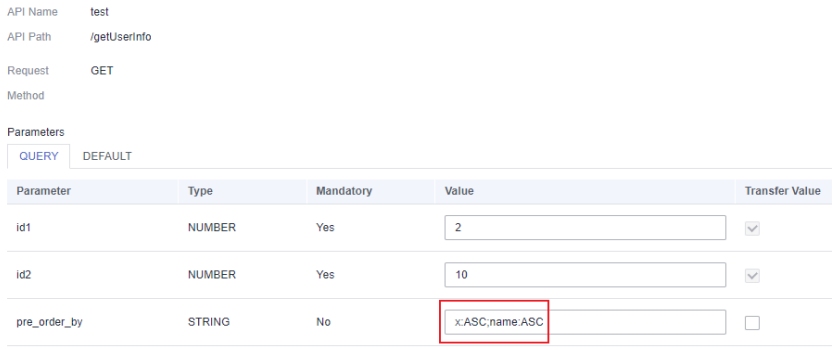
2. (Optional) Change the value of **pre_order_by**, which indicates the ranking parameter description.

   The system provides a default value based on the ranking parameter value, which indicates the ascending order. Generally, the value of **pre_order_by** is in either of the following formats: ***Ranking parameter name***:ASC (ascending order) or ***Ranking parameter name***:DESC (descending order). Separate multiple ranking parameter descriptions by semicolons (;).

   NOTE

   - The **pre_order_by** parameter is optional. By default, the default value (ascending) of the mandatory ranking field is used.
   - Ensure that you set the **pre_order_by** parameter by strictly following the ranking parameter sequence, optional attributes, and ranking mode configured in the ranking parameter list. Otherwise, the API cannot be called.

Figure 10-21 Changing the value of pre_order_by



3. (Optional) Change the values of pagination parameters.

   The system displays the returned data on multiple pages. Parameter **pageSize** indicates the size of a page, and **pageNum** indicates the page number. By default, the page size is 100, and data on the first page is returned.

> **NOTE**
>
> During API debugging, the maximum value of **page_size** is **100**. If the value of **page_size** is greater than **100**, 100 records are returned by default.

Figure 10-22 Changing the values of pagination parameters

4. After setting and saving all parameters, click **Next**.

   Specify **Value** and click **Debug**. You can view the **Request** and **Response** details on the right part of the displayed page.

   - During the test, if the DataArts DataService API does not return a query result within 30 seconds (default value), a timeout error is reported.
   - If the test fails, follow the instructions as prompted and restart the test.

After the test is complete, click **OK**.

Modifying the API

To modify an API, choose **API Development** > **API Catalogs** or **API Development** > **APIs**, locate the API, and click **Edit** to modify the API.

> **NOTE**
>
> An API can be edited only when it is in the **Created**, **Rejected**, **Offline**, or **Disabled** state.

10.3.2.2 Generating an API Using a Script or MyBatis

This section describes how to generate an API using a script or MyBatis.

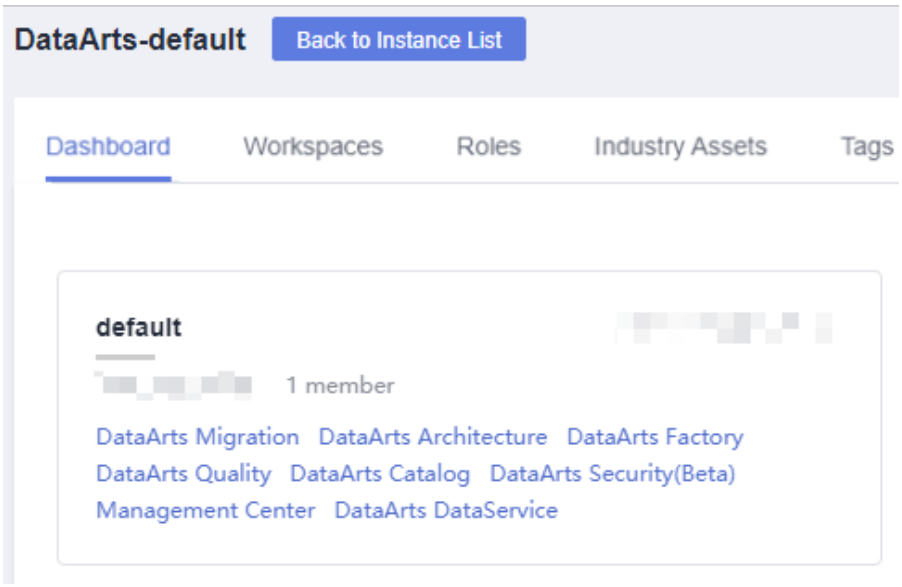
This mode can meet personalized query requirements of users. It allows you to compile API query SQL statements and provides multi-table join, complex query conditions, aggregation functions, and more capabilities.

- Script: Only common SQL syntax is supported.
- MyBatis: Only DataArts DataService Exclusive supports this mode. In this mode, the script supports the Mybatis tag syntax. The parameter parsing format is #{parameter}. Tag syntax such as if, choose, when, foreach, and where is supported. You can use the tag syntax to implement complex query logic such as null value verification, multi-value traversal, dynamic table query, dynamic sorting, and aggregation.

Configuring Basic API Information

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

   Figure 10-23 DataArts DataService

2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **APIs** from the left navigation bar, and click **Create**. On the displayed page, enter the basic information.

   Table 10-8 API basic configuration

Parameter	Description
API	An API name consists of 3 to 64 characters and starts with a letter. Only letters, numbers, and underscores (_) are allowed.
API Catalog	A collection of APIs for a specific function or scenario. It is similar to a folder and specifies the location of APIs. You can search for APIs in a specified API catalog. The API catalog is the minimum organization unit of APIs in DataArts DataService and also the minimum management unit in the API gateway. Click **Select Catalog** to create an API catalog or select an existing one created in **Creating an API Directory**.

Parameter	Description
Request Path	API access path, for example, **/getUserInfo** It is the part between the domain name and query parameters in the URL of a request path, for example, **/blogs/*xxxx*** shown in the following figure. **Figure 10-24** API access path in the URL  Braces ({}) can be used to identify parameters in a request path as wildcard characters. For example, **/blogs/{blog_id}** indicates that any parameter can follow **/blogs**. **/blogs/188138** and **/blogs/0** can both match **/blogs/{blog_id}**, and are processed by this API. In addition, duplicate request paths are not allowed for the same domain name. When a path parameter is used as a wildcard, the name is not unique. For example, **/blogs/{blog_id}** and **/blogs/{xxxx}** are considered as the same path.
Parameter Protocol	Protocol used to transmit requests. The shared edition supports HTTP and HTTPS, and the exclusive edition supports HTTPS. HTTPS is recommended. HTTP is insecure and may have security risks. • HTTP is a basic network transmission protocol. It is stateless, connectionless, simple, fast, and flexible, and uses plaintext for transmission. It is easy to use but has poor security. • HTTPS is an HTTP-based protocol with SSL or TLS encryption verification. It can effectively verify identities and protect data integrity. To access HTTPS APIs, you need to configure related SSL certificates or skip SSL verification.
Request Method	HTTP request method, indicating the type of the requested operation, such as GET and POST. The method complies with the resultful style. • **GET** requests the server to return specified resources. This method is recommended. • **POST** requests the server to add resources or perform special operations. This method is used only for API registration. The POST request does not have a body. Instead, it involves transparent transmission.
Description	A brief description of the API to create.

Parameter	Description
Tag	API tag. The tag is used to mark the API attributes. After the API is created, you can quickly search for the API by tag. A maximum of 20 tags can be set for an API.
Reviewer	A reviewer who has permissions to review APIs. Click **Add** to enter the **Review Center** page and click **Add** on the **Reviewers** tab page to add a reviewer.
Security Authentication	When creating an API, you can select one of the following authentication modes. The three modes differ in how the API is called. You are advised to use **App Authentication**, which is more secure that the other two modes. • **App authentication**: App authentication is used for calling an API. The AppKey & AppSecret is used for authentication. It is highly secure. When **App authentication** is used, an SDK is required for access. Java, Go, Python, JavaScript, C#, PHP, C++, C, and Android SDKs are available. For details about how to call APIs in each language, see **Calling APIs Through App Authentication**. • **IAM authentication**: IAM authenticates API requests. This mode is available only for Huawei cloud users. The security level is medium. When using IAM authentication, you need to call the **Obtaining a User Token** API of IAM to obtain a token, add the **X-Auth-Token** parameter with the obtained token as the value to the request header, and use an API calling tool or SDK to call released APIs. • **Non-authentication**: No authentication is required. This mode allows all users to access APIs, which may pose security risks. It is recommended only for testing APIs. If the caller is not a trusted user, there is a risk of data leakage, breakdowns caused by high concurrent access, SQL injection, and others. This mode does not require any authentication information. You can use an API calling tool or SDK to directly call an API by specifying required parameters.
Display Scope	After the API is published, all users in the selected scope can view the API in the service catalog. • Current workspace APIs • Current project APIs • Current tenant's APIs
Access Log	If you select this option, the API query result will be recorded and retained for seven days. You can choose **Operations Management** > **Access Logs** and select the request date to view the logs.

Parameter	Description
Min. Retention Period	Minimum retention period of the API publishing status, in hours. Value **0** indicates that the retention period is not limited. You can suspend, unpublish, or cancel authorization for an API only after the minimum retention period ends. The system notifies the authorized users. If all authorized users have processed the notifications or unbound the API from their apps, the API will be suspended or unpublished, or the API authorization will be canceled. Otherwise, the system will forcibly suspend, unpublish, or cancel authorization for the API when the minimum retention periods ends. For example, if the minimum retention period is set to 24 hours, the API can be suspended 24 hours after it is published. If the authorized user handles the notifications in the review center or unbind the API from the app, the API will be directly suspended. Otherwise, the API will be forcibly suspended when the minimum retention period ends.

Parameter	Description
Input Parameters	Parameters required for calling the API. The parameters are used as the request parameters on the **Set Data Extract Logic** page. An input parameter consists of the parameter location, parameter type, whether the parameter is mandatory, and the default value. ● The parameter location can be **Query**, **Header**, **Path**, or **Body**. In addition, static parameters are supported. – **Query** is the query parameter following the URL. It starts with a question mark (?) and connects multiple parameters with **&**. – **Header** is located in the request header and is used to transfer current information, for example, host and token. – **Path** is a request parameter in the request path. If you configure a path parameter, you must also add this parameter to the request path. – **Body** is a parameter in the request body and is generally in JSON format. – **Static** is a static parameter that does not change with the value passed by API callers. The parameter value is determined upon API authorization. If the parameter value is not set during authorization, the default value of the API input parameter is used. ● The parameter type can be **Number** or **String**. **Number** corresponds to numeric data types such as int, double, and long. **String** corresponds to text data types such as char, vachar, and text. ● **Mandatory** and **Default Value**: If you select **Yes** for **Mandatory**, parameters must be passed for accessing the API. Otherwise, the default value of the parameter will be used if the parameter is not passed for accessing the API. **NOTE** When defining an input parameter, ensure that the following size requirements are met: ● **Query** and **Path**: 32 KB. ● **HEADER**: The maximum size is 128 KB. ● **BODY**: The maximum size is 128 KB. You need to set input parameters based on the designed request parameters for the API. For example, the request path of the API used to query user information in a table by user ID is **/getUserInfo**. You can configure input parameters as follows: ● If the request parameter for calling the API is **id**, and the information about the user with **id** needs to be returned, configure an input parameter as follows:

Parameter	Description
	1. Click **Add** and enter **id** for **Name**. 2. Set **Parameter Location** to **Query**. 3. Set **Type** to **Number**. 4. Select **Yes** for **Mandatory**. 5. Retain the default value. • If the request parameters for calling the API are **id1** and **id2**, and the user information between **id1** and **id2** needs to be returned, configure input parameters as follows: 1. Click **Add** and enter **id1** for **Name**. 2. Set **Parameter Location** to **Query**. 3. Set **Type** to **Number**. 4. Select **Yes** for **Mandatory**. 5. Retain the default value. 6. Click **Add** again and configure parameter **id2**.

4. After the basic API information is complete, click **Next** to go to the **Data Extract Logic** page.

Configuring the Data Extraction Logic

NOTE

This section uses a script to describe how to configure the API data extraction logic. The configuration procedure for the Mybatis mode is the same as that for the script mode, except for the parameter parsing mode and supported syntax.

If you use Mybatis to generate an API, you need to change the parameter parsing format in the scripts in this section from ***${parameter}*** to ***#{parameter}***. In addition, you can click in the script editing area to view the tag syntax supported by Mybatis.

Set **Data Acquisition Method** to **Script** or **MyBatis**.

1. Set **Data Source**, **Data Connection**, and **Database**.

   NOTE

   For details on the data sources supported by DataArts DataService, see **Data Sources**. Configure data sources in Management Center in advance and enter SQL statements as prompted.

2. Set **Paging Mode**. You are advised to select **Custom**.
   - Default pagination: If you enter a SQL script when creating an API, DataArts DataService automatically adds the pagination logic to the SQL script.

     For example, if you enter the following SQL script:

     ```
     SELECT * FROM userinfo WHERE id=${userid}
     ```

     When processing API debugging or calling, DataArts DataService automatically adds the pagination logic to the preceding SQL script and generates the following script:

     ```
     SELECT * FROM (SELECT * FROM userinfo WHERE id=${userid}) LIMIT {limitValue} OFFSET {offsetValue}
     ```


limitValue indicates the number of data records to be read, and **offsetValue** indicates the number of data records to be skipped (that is, offset). For example, if **limitValue** is set to **20** and **offsetValue** is set to **40**, 40 data records will be skipped and 20 will be read. Generally, **limitValue** and **offsetValue** can be used as input parameters. You can transfer values to them when debugging or calling an API. If **limitValue** or **offsetValue** is not specified, the system assigns default values to them.

- Custom pagination: DataArts DataService does not process the SQL script for creating an API. You need to define the pagination logic when writing the SQL statement.

  If **limitValue** (number of data records to be read) and **offsetValue** (number of data records to be skipped) have been obtained, you can define the pagination logic using the following script:

  ```
  SELECT * FROM userinfo WHERE id=${userid} LIMIT {limitValue} OFFSET {offsetValue}
  ```

  More commonly, you can use **pageSize** and **pageNum** to define the pagination logic. The script format is as follows:

  ```
  SELECT * FROM userinfo WHERE id=${userid} LIMIT {pageSize} OFFSET {pageSize*(pageNum-1)}
  ```

  NOTE

  The syntax style varies depending on the data source, and so does the pagination script. The following are some example data sources:

  - DLI does not support the **LIMIT {limitValue} OFFSET {offsetValue}** format. It only supports the **LIMIT {limitValue}** format.
  - HetuEngine does support the **LIMIT {limitValue} OFFSET {offsetValue}** format. It only supports the **OFFSET {offsetValue} LIMIT {limitValue}** format.

3. Compile the SQL statement for a query API.

   On the script editing page, click next to **Edit Script** and develop a SQL query statement as prompted. You can click to add input parameters to the SQL statement as API request parameters.

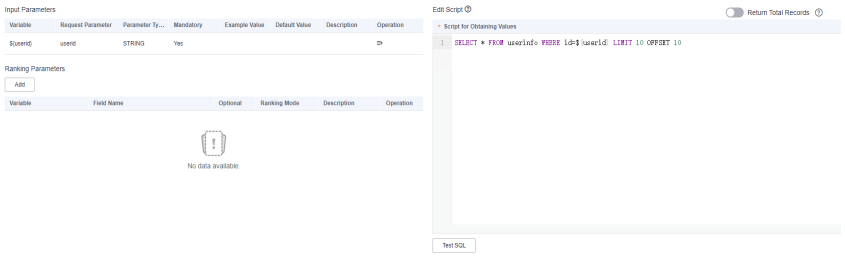
   For example, you can write the following script to query user information in a user table based on the user ID. **id** is a field in the **userinfo** table, and **userid** is an input parameter defined for the API.

   ```
   SELECT * FROM userinfo WHERE id=${userid}
   ```

   If custom pagination is used, the value of **pageSize** is **10**, and the value of **pageNum** is **2**, write the following script based on the **LIMIT {pageSize} OFFSET {pageSize*(pageNum-1)}** conversion method:

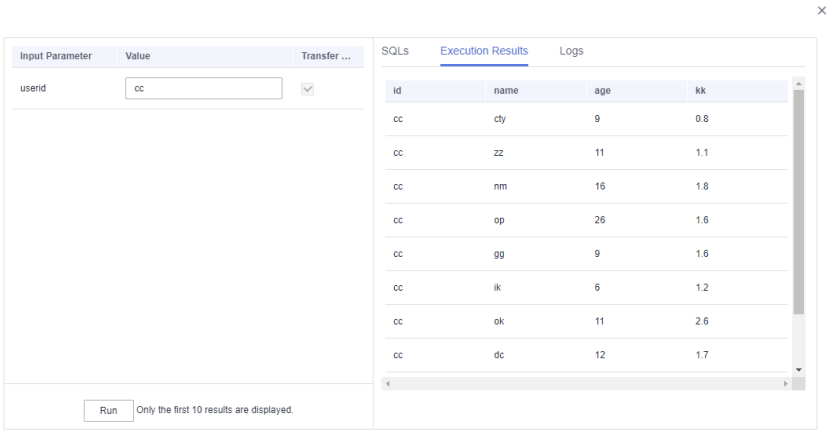
   ```
   SELECT * FROM userinfo WHERE id=${userid} LIMIT 10 OFFSET 10
   ```

   Figure 10-25 Compiling the SQL statement for a query API

Click **Test SQL** under the script editing window, set the value for the input parameter, and click **Run** to check whether the expected result can be returned. If the test fails, you can check whether the SQL statement meets the expectation on the **SQLs** tab page or view the error message on the **Logs** tab page.

Figure 10-26 Testing the SQL statement



NOTE

- The fields obtained by SELECT are the response parameters of the API. (The aliases can be obtained through AS.)
- Parameters in the where condition are API request parameters. In the script mode, the parameter format is **${Parameter name}**. In the MyBatis mode, the parameter format is **#{Parameter name}**.
- You can enable **Return Total Records**. Then the total number of script execution results will be returned.
- If you want to set multiple values for a parameter, observe the following format:
  - String: 'a','b','c'
  - Value: 1,2
  - Field: a,b,c

4. Add ranking parameters.

   In the ranking parameter list, click **Add** to add ranking fields.

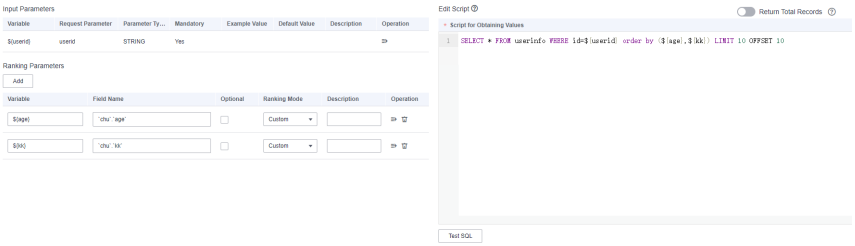
   - Field names are invisible to external systems. They are fields of the selected tables and are accessed during an API call. If the query SQL statement of the API has been compiled and verified through a test, you can select a ranking field from the **Field Name** text box.
   - Variables can be customized and associated with field names. Enter a parameter name in the **Variable** text box. The system automatically changes the parameter name to a variable.
   - If you select **Optional**, the parameter is optional.
   - The ranking mode can be ascending, descending, or custom. If you set **Ranking Mode** to **Ascending** or **Descending**, but set **pre_order_by** to a value different from the value of **Ranking Mode** when testing or calling the API, the API cannot be called.

Ranking parameters take effect only after they are added to the SQL script. Click ⇉ to add ranking parameters to the SQL statement and use **ORDER BY** to sort the parameters.

For example, you can write the following script to query user information in a user table based on the user ID, with **age** and **kk** used to sort the query results and **pageSize** and **pageNum** set to **10** and **2**, respectively.

```
SELECT * FROM userinfo WHERE id=${userid} order by (${age},${kk}) LIMIT 10 OFFSET 10
```

Figure 10-27 Adding ranking parameters



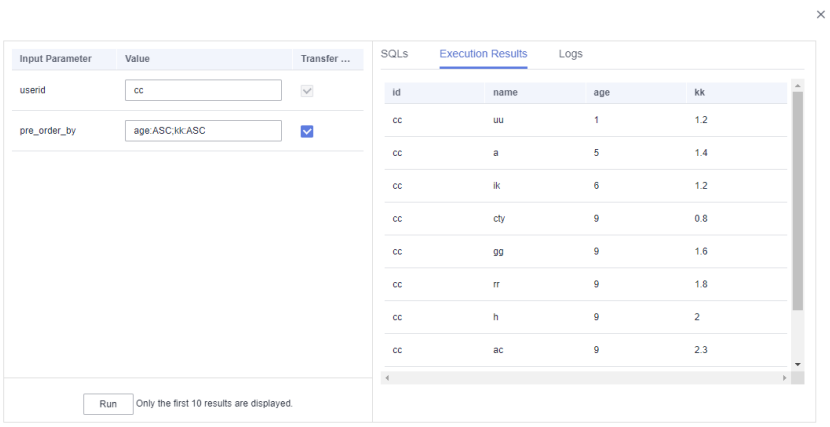
Click **Test SQL** under the script editing window, set the values for **userid** and **pre_order_by**, and click **Run** to check whether the expected result can be returned. The default value of **pre_order_by** is provided by the system based on the ranking parameter information, which is the ascending order. Generally, the value of **pre_order_by** is in either of the following formats: ***Ranking parameter name***:ASC (ascending order) or ***Ranking parameter name***:DESC (descending order). Separate multiple ranking parameter descriptions by semicolons (;).

If the test fails, you can check whether the SQL statement meets the expectation on the **SQLs** tab page or view the error message on the **Logs** tab page.

NOTE

- The **pre_order_by** parameter is optional. By default, the default value (ascending) of the mandatory ranking field is used.
- Ensure that you set the **pre_order_by** parameter by strictly following the ranking parameter sequence, optional attributes, and ranking mode configured in the ranking parameter list. Otherwise, the API cannot be called.

Figure 10-28 Testing the SQL statement



5. Click **Next** to go to the API test page.

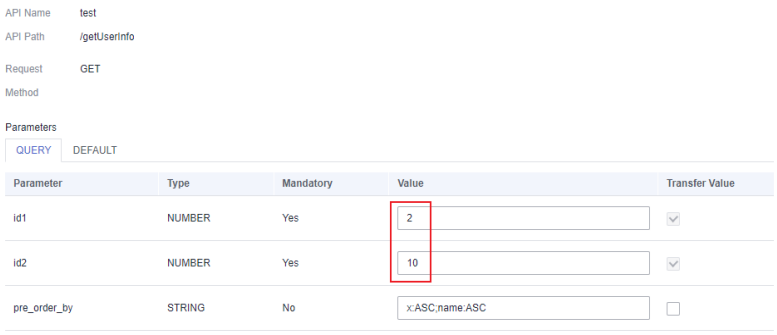
Testing the API

1. Set values for input parameters.

   If you want to set multiple values for a parameter, observe the following format:

   - String: 'a','b','c'
   - Value: 1,2
   - Field: a,b,c

   Figure 10-29 Setting values for input parameters



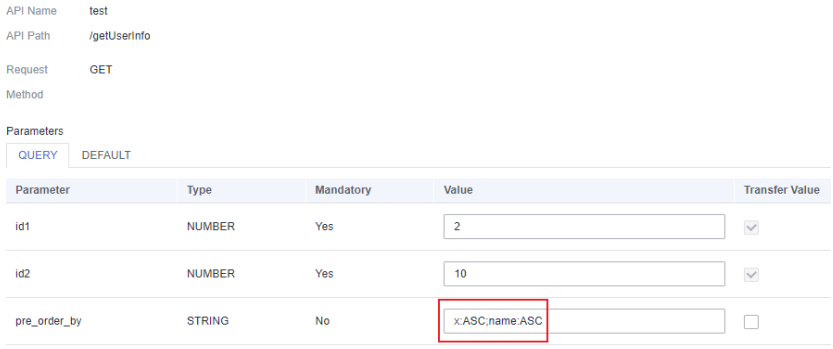
2. (Optional) Change the value of **pre_order_by**, which indicates the ranking parameter description.

   The system provides a default value based on the ranking parameter value, which indicates the ascending order. Generally, the value of **pre_order_by** is in either of the following formats: ***Ranking parameter name***:ASC (ascending order) or ***Ranking parameter name***:DESC (descending order). Separate multiple ranking parameter descriptions by semicolons (;).

   NOTE

   - The **pre_order_by** parameter is optional. By default, the default value (ascending) of the mandatory ranking field is used.
   - Ensure that you set the **pre_order_by** parameter by strictly following the ranking parameter sequence, optional attributes, and ranking mode configured in the ranking parameter list. Otherwise, the API cannot be called.

Figure 10-30 Changing the value of pre_order_by



3. (Optional) View the values of pagination parameters.

   If the default pagination mode is used, you can view the pagination parameters. **pageSize** indicates the size of a page, and **pageNum** indicates the page number. By default, the page size is 100, and data on the first page is returned.

   Figure 10-31 View the values of pagination parameters

4. After setting and saving all parameters, click **Next**.

   Specify **Value** and click **Debug**. You can view the **Request** and **Response** details on the right part of the displayed page.

   - During the test, if the DataArts DataService API does not return a query result within 30 seconds (default value), a timeout error is reported.
   - If the test fails, follow the instructions as prompted and restart the test.

After the test is complete, click **OK**.

Modifying the API

To modify an API, choose **API Development** > **API Catalogs** or **API Development** > **APIs**, locate the API, and click **Edit** to modify the API.

NOTE

An API can be edited only when it is in the **Created**, **Rejected**, **Offline**, or **Disabled** state.

10.3.2.3 Registering APIs

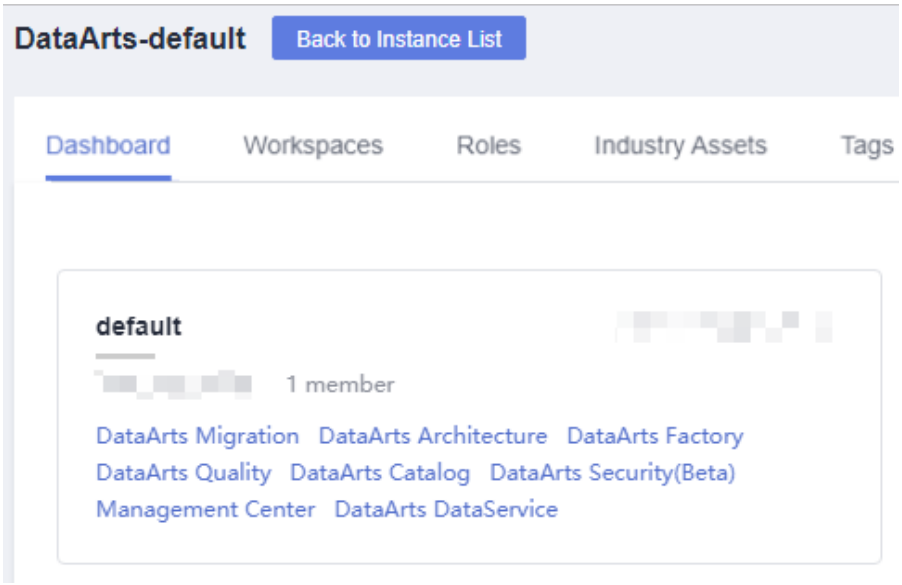
This document describes how to use DataArts DataService to register existing service APIs with API Gateway, publish these APIs, and centrally manage these APIs and the APIs created in DataArts DataService.

DataArts DataService Shared supports the registration of RESTful APIs using GET and POST methods.

Configuring Basic API Information

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-32 DataArts DataService



2. In the left navigation pane, choose **DataArts DataService Shared**. The **Overview** page is displayed.
3. In the left navigation pane, choose **API Development** > **APIs**. In the right pane, click **Register** and configure basic API information.

Table 10-9 API basic configuration

Parameter	Description
API	An API name consists of 3 to 64 characters and starts with a letter. Only letters, numbers, and underscores (_) are allowed.
API Catalog	An API catalog is a set of APIs with a specific function or scenario. It is the minimum organization unit of APIs in DataArts DataService and the minimum management unit of API Gateway. Click **Select Catalog** to create an API catalog or select an existing one created in **Creating an API Directory**.

Parameter	Description
Request Path	The path for accessing an API. Example: **/v2/{project_id}/streams**.
Protocol	A protocol used to transmit requests. HTTP and HTTPS are supported.
Request Mode	HTTP defines the following request modes that can be used to send a request to the server. **GET** requests the server to return specified resources. **POST** requests the server to add resources or perform special operations. This method is recommended for API registration. The POST request does not have a body. Instead, it involves transparent transmission.
Description	A brief description of the API to be registered.
Tag	The name of the tag. Only letters, numbers, and underscores (_) are allowed. Tag names cannot start with underscores (_).
Reviewer	An owner who has permissions to review APIs. Click **Add** to enter the **Review Center** page. On the page displayed, click **Add** on the **Reviewer Management** tab page to add a reviewer.

Parameter	Description
Security Authentication	When creating an API, you can select one of the following authentication modes. The three modes differ in how the API is called. You are advised to use **App Authentication**, which is more secure that the other two modes. • **App authentication**: App authentication is used for calling an API. The AppKey & AppSecret is used for authentication. It is highly secure. When **App authentication** is used, an SDK is required for access. Java, Go, Python, JavaScript, C#, PHP, C++, C, and Android SDKs are available. For details about how to call APIs in each language, see **Calling APIs Through App Authentication**. • **IAM authentication**: IAM authenticates API requests. This mode is available only for Huawei cloud users. The security level is medium. When using IAM authentication, you need to call the **Obtaining a User Token** API of IAM to obtain a token, add the **X-Auth-Token** parameter with the obtained token as the value to the request header, and use an API calling tool or SDK to call released APIs. • **Non-authentication**: No authentication is required. This mode allows all users to access APIs, which may pose security risks. It is recommended only for testing APIs. If the caller is not a trusted user, there is a risk of data leakage, breakdowns caused by high concurrent access, SQL injection, and others. This mode does not require any authentication information. You can use an API calling tool or SDK to directly call an API by specifying required parameters.
Display Scope	After the API is published, all users in the selected scope can view the API in the service catalog. • Current workspace APIs • Current project APIs • Current tenant's APIs
Access Log	If you select this option, the API query result will be recorded and retained for seven days. You can choose **Operations Management** > **Access Logs** and select the request date to view the logs.
Min. Retention Period	Minimum duration reserved before API unbinding. Before an API developer suspends, unpublishes, or cancels the authorization of an API, the system notifies the authorized API callers and reserves at least *X* hours for them to unbind the API. During the retention period, the API can be used if it is not unbound. The value **0** indicates that there is no minimum retention period.

Parameter	Description
Input Parameter	Input parameter is a set of parameters in the API request, including dynamic parameters in the resource path, query parameters in the request URI, and header parameters. The following is an example that describes the dynamic parameters in the resource path (request path): **/v2/{project_id}/streams**, where **{project_id}** is a dynamic parameter that needs to be configured. 1. Click **Add** and enter **project_id** for **Name**. 2. Set **Parameter Location** to **PATH**. 3. Set **Type** to **STRING**. 4. Set **Example Value** and **Description** as required.

4. After the basic API information is complete, click **Next** to go to the **Data Extract Logic** page.

Configuring API Parameters

After configuring basic API information, you can set API parameters. The following describes how to configure the API backend services and request parameters.

Table 10-10 API parameters

Parameter	Description
Protocol	A protocol used to transmit requests. HTTP and HTTPS are supported. This parameter is used by DataArts DataService to transmit requests to the APIs to be registered.
Request Mode	HTTP defines the following request modes that can be used to send a request to the server. This parameter is used by DataArts DataService to transmit requests to the APIs to be registered. **GET** requests the server to return specified resources. **POST** requests the server to add resources or perform special operations.
Backend Service Host	Host of the API to be registered. The value cannot start with **http://** or **https://** and cannot contain **Path**.
Backend Service Path	Path of the API to be registered. The path can contain parameters placed in {}, for example, **/user/{*userid*}**.
Backend Timeout (ms)	Backend timeout interval.

Parameter	Description
Backend Service Parameter	The optional parameters can be placed in **PATH**, **Header**, and **Query**. The positions of optional parameters vary depending on the request mode. Select a parameter position as required.
Constant Parameter	Constant parameter is the fixed parameter invisible to the caller. Constant parameter does not need to be transferred during API calling. However, the background service always receives the constant parameter and parameter value defined here. This parameter applies to scenarios in which you want to set a parameter of an API to a fixed value and hide the parameter from the caller.

Testing an API

After setting and saving all parameters, click **Next**.

Specify **Value** and click **Debug**. You can view the **Request** and **Response** details on the right part of the displayed page.

- During the test, if the DataArts DataService API does not return a query result within 30 seconds (default value), a timeout error is reported.
- If the test fails, follow the instructions as prompted and restart the test.

After the test is complete, click **OK**.

10.3.3 Debugging an API

Scenarios

You can debug an API on the management console by adding HTTP header parameters and body parameters.

NOTE

- APIs whose backend paths contain environment variables cannot be debugged.
- APIs bound to a signature key cannot be debugged.
- If a request throttling policy has been bound to an API, the policy does not take effect when you debug the API.

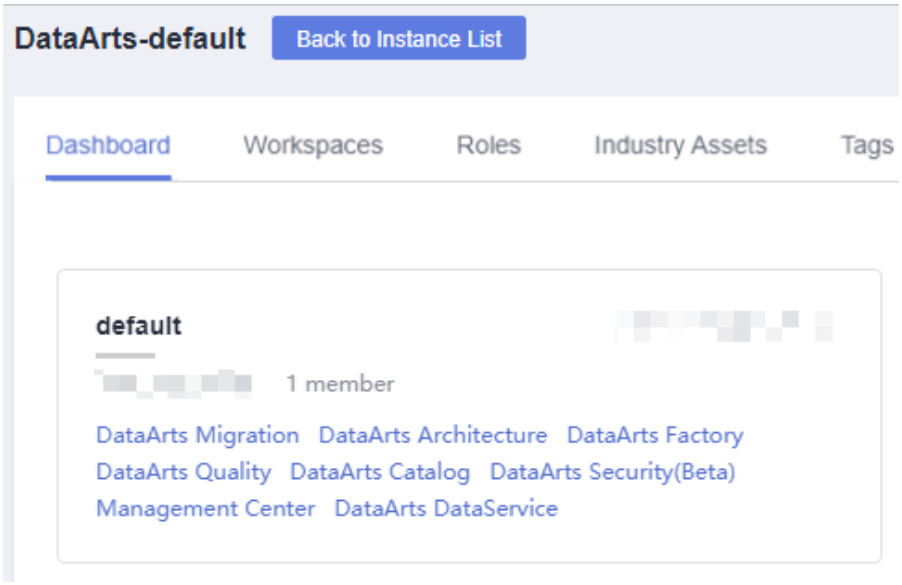
Prerequisites

- An API has been created.
- The backend service has been set up.

Procedure

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-33 DataArts DataService



2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **APIs**.
4. Use either of the following methods to debug an API:
   - Locate the row that contains the target API, and choose **More** > **Debug**.
   - Click the name of the target API, and click **Test** on the displayed API details page.

     You can configure API request parameters in the left pane. See **Table 10-11** for parameter details. The request information sent by the API and the returned result after the API request is invoked are displayed on the right.

     Table 10-11 Debugging APIs

Parameter	Description
Parameters	Query parameters and their values.
Cluster Settings	Supported only by Exclusive Edition. Select the instance where the API to be debugged resides.

     NOTE

     The information displayed on the debugging page varies according to the request type.
5. After request parameters are added, click **Debug**.

   The API calling response information is displayed in the command output area in the right pane.

- If the API is successfully called, HTTP status code 200 and response information are returned.
- If no result is returned within 30 seconds (default value), a timeout error is reported.
- If the debugging fails, the HTTP status code 4*xx* or 5*xx* is returned.

6. You can send different requests using varied parameters and values to verify the API.

   NOTE

   To modify the API parameters, click **Edit** in the upper right corner. The API editing page is displayed.

Follow-up Procedures

After the API is successfully debugged, you can publish the API so that the API can be called by users. For how to publish an API, see **Publishing an API**.

10.3.4 Publishing an API

This section describes how to publish an API to service catalogs.

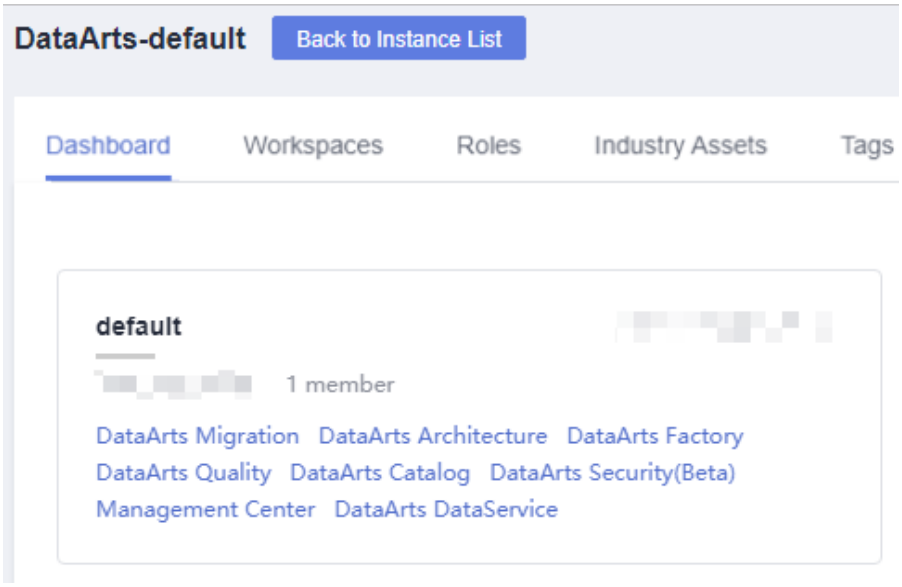
Scenario

For the sake of security, APIs generated and registered in DataArts DataService must be published to service catalogs before they can provide services.

Procedure

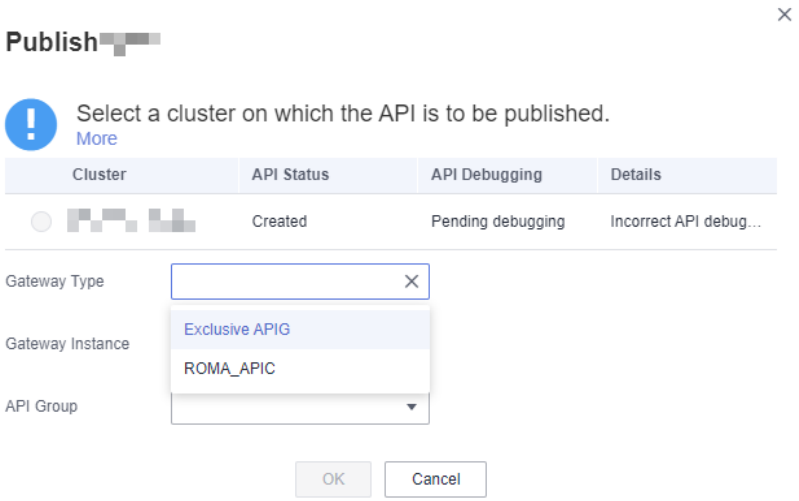
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

   Figure 10-34 DataArts DataService



2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. In the navigation pane, choose **API Development** > **APIs**. Locate an API, click **More** in the **Operation** column, and select **Publish**.
4. In the displayed dialog box, you can click **More** to view details.

Figure 10-35 Publish



- In DataArts DataService Exclusive, the API is published to a DataArts DataService Exclusive cluster by default. After the API is published, it can be called through the intranet. Only the APIs published to a DataArts DataService Exclusive cluster can be called for an unlimited number of times. Otherwise, an API can be called 1,000 times at most every day.

  You can also click **More** and select an APIG Exclusive or ROMA Connect instance to publish the API to.

  - APIG Exclusive: To publish an API to APIG Exclusive, you must buy an APIG instance on the APIG console in advance. After the instance is created, a default API group is available. The system automatically assigns a debugging domain name for internal tests to the API group. This debugging domain name is unique and cannot be changed, and it can be accessed for a maximum of 1,000 times each day. If you want to create an API group exclusively for DataArts DataService APIs, see **Creating an API Group**. In addition, you can bind one or more independent domain names to an API group. For details, see **Binding a Domain Name**. The domain names can be used to call APIs for more than 1,000 times each day.
  - ROMA Connect instance: To publish an API to a ROMA Connect instance, you must create a ROMA Connect instance and an API group on the ROMA Connect console in advance. For details, see **Creating an API Group**. The system automatically allocates a subdomain name to the API group for internal testing. The subdomain name can be accessed for a maximum of 1,000 times each day. You can also bind independent domain names to the API

group so that they can be used to call your published APIs. For details, see **Binding Domain Names**.

5. APIs must be reviewed and approved before they can be published. APIs are reviewed in the following way:
   - An API publisher who does not have the reviewer permission must submit the API to the reviewer for review.
   - An API publisher who has the reviewer permission can publish an API without review or approval.

   An API submitted by a non-reviewer is published after it is approved by the reviewer.

   NOTE

   The data connection of an API in the pending review state cannot be changed. It can be changed only when the application is rejected by a user with the workspace administrator role.

   An admin, developer, or operator can be a reviewer. A viewer cannot be a reviewer.

   Regardless of whether they are added as reviewers, users with the admin role of the workspace have the reviewer permissions by default.

Follow-up Operations

After the API is published, you can go to the **Service Catalogs** page to view the API information.

You can also manage APIs. For details, see **Managing APIs**. Alternatively, you can choose **Operations Management** > **Throttling Policies** and configure throttling for the API. For details, see **Creating Throttling Policies**.

10.3.5 Managing APIs

10.3.5.1 Displaying an API

Scenario

If you want to change the visibility scope of an API in the service catalog, you can use the **Display** function or set the **Display Scope** parameter for the API.

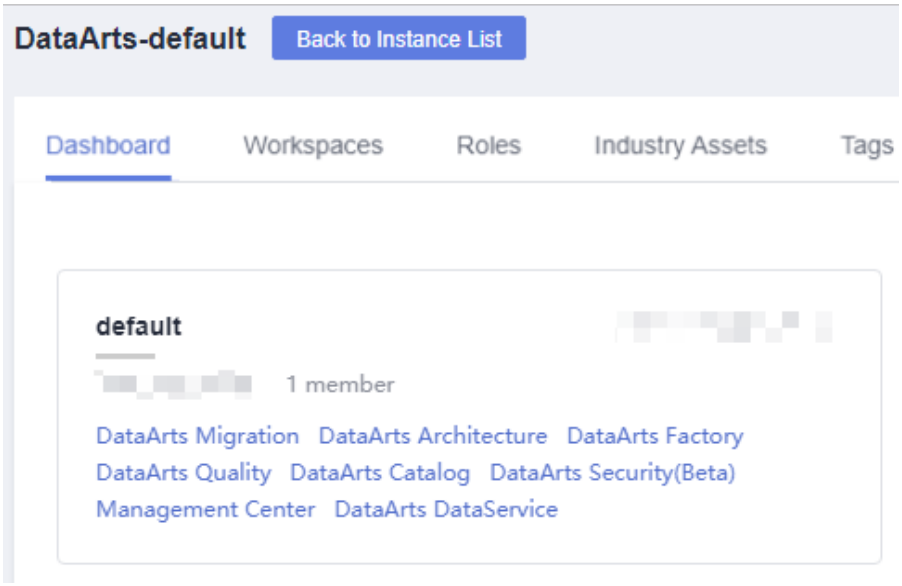
Prerequisites

An API has been created.

Changing the API Visibility Scope Using the Display Function

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

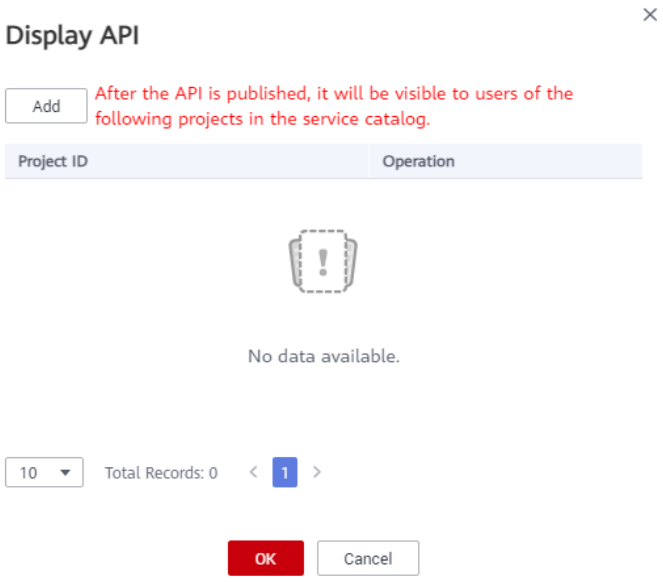
Figure 10-36 DataArts DataService



1. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
2. Choose **API Development** > **API Catalogs** or **API Development** > **APIs**. Locate an API, click **More** in the **Operation** column, and select **Display**.
3. In the displayed dialog box, click **Add**, enter a project ID, and click **OK** to make the API visible to users in the project.

   For how to obtain the project ID, see **(Optional) Obtaining Authentication Information**.

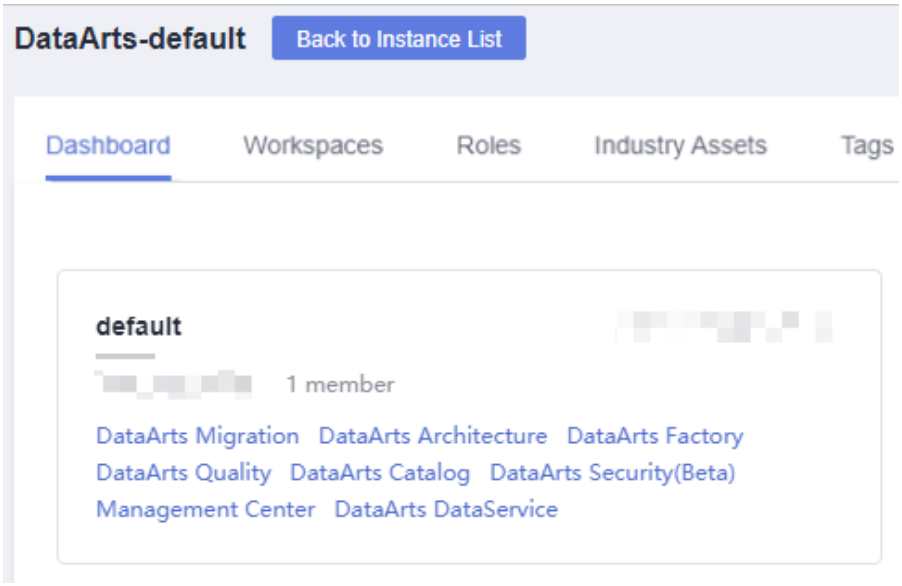
Figure 10-37 Display API



Changing the API Visibility Scope by Setting the Display Scope Parameter

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

   Figure 10-38 DataArts DataService

2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **API Catalogs** or **API Development** > **APIs**. Locate an API and click **Edit** in the **Operation** column. An API can be edited only when it is in the **Created**, **Rejected**, **Offline**, or **Disabled** state.
4. On the **Configure Basic Details** page, select a value for the **Display Scope** parameter. The value can be **Current workspace's APIs**, **Current project's APIs**, or **Current tenant's APIs**. Then save the modification.
5. Restore or publish the API again to change the visibility scope of the API in the service catalog.

10.3.5.2 Suspending/Restoring an API

Scenarios

To edit or debug a published API, you must suspend the API first. After the API is suspended, its original authorization information is retained. You can edit and debug the API.

You can restore the API so that it can continue to provide services.

NOTE

The suspended API cannot be accessed in the specified time, which may affect the applications or users who are using the API. Ensure that users have been notified of this consequence.

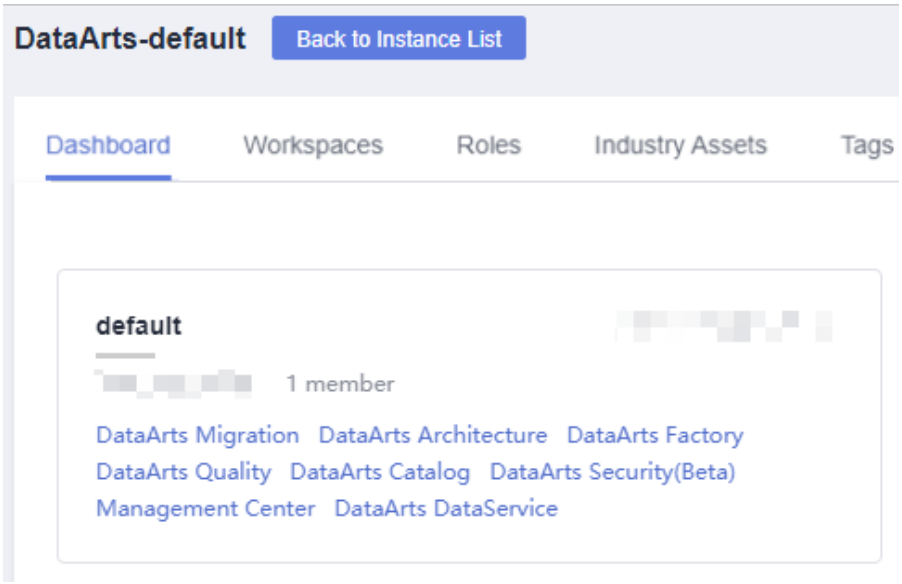
Prerequisites

- An API has been created.
- An API has been published in the environment.

Suspending an API

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

   Figure 10-39 DataArts DataService

2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **APIs**.
4. Locate the row that contains the API to be suspended, click **More** in the **Operation** column, and select **Suspend**.
5. In the displayed dialog box, select the time period when the API needs to be suspended and click **OK**.

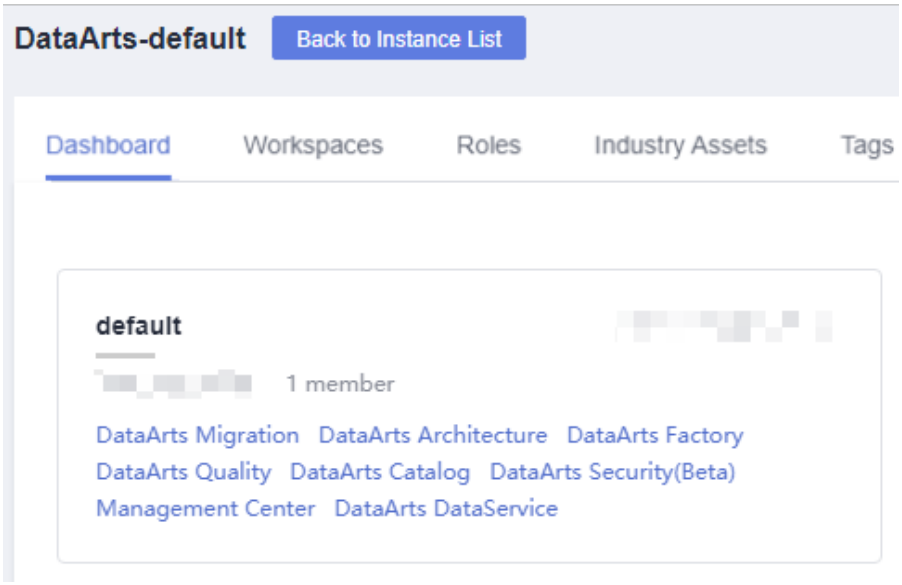
   NOTE

   The API suspension time must be later than its minimum retention period. Authorized users will be notified of the suspension. If all authorized users process the notifications in the review center or unbind the API from their apps, the API will be directly suspended. Otherwise, the API will be forcibly suspended when the minimum retention period ends.

Restoring an API

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-40 DataArts DataService



2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Locate the row that contains the API to be restored, click **More** in the **Operation** column, and select **Restore**.

10.3.5.3 Unpublishing/Deleting APIs

Scenario

If you want to stop an API that has been published from providing services, you can unpublish the API. For details, see **Unpublishing an API**.

- If you want to continue to use an API that has been unpublished, you need to publish it again. Note that the original authorization information of the API will not be retained once the API is unpublished.
- If you no longer need the API, you can delete it. For details, see **Deleting APIs**.

NOTE

The unpublished API cannot be accessed in the specified time, which may affect the applications or users who are using the API. Ensure that users have been notified of this consequence.

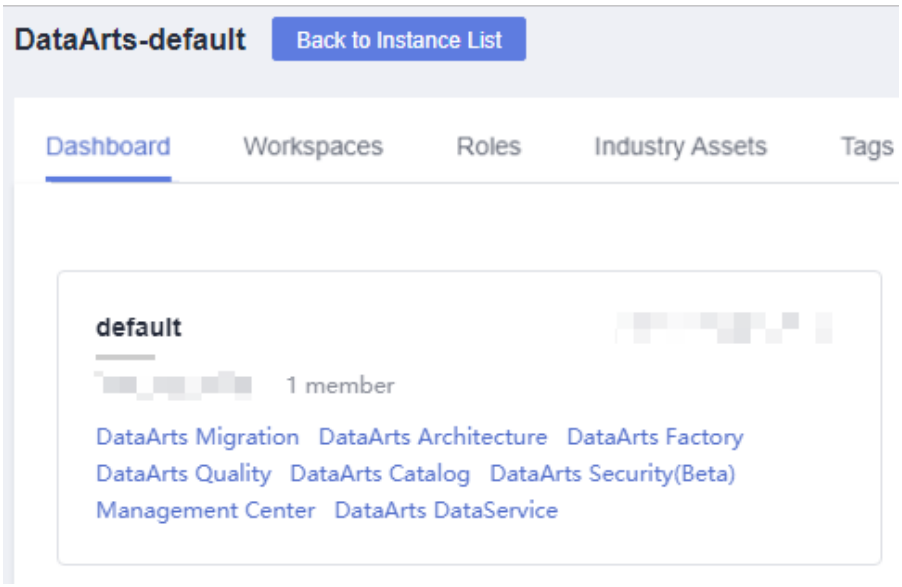
Prerequisites

- An API has been created.
- The API has been published.

Unpublishing an API

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-41 DataArts DataService



2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **APIs**.
4. Locate the row that contains the target API, choose **More** > **Unpublish**.
5. In the displayed dialog box, select the time period where the API needs to be unpublished and click **OK**.

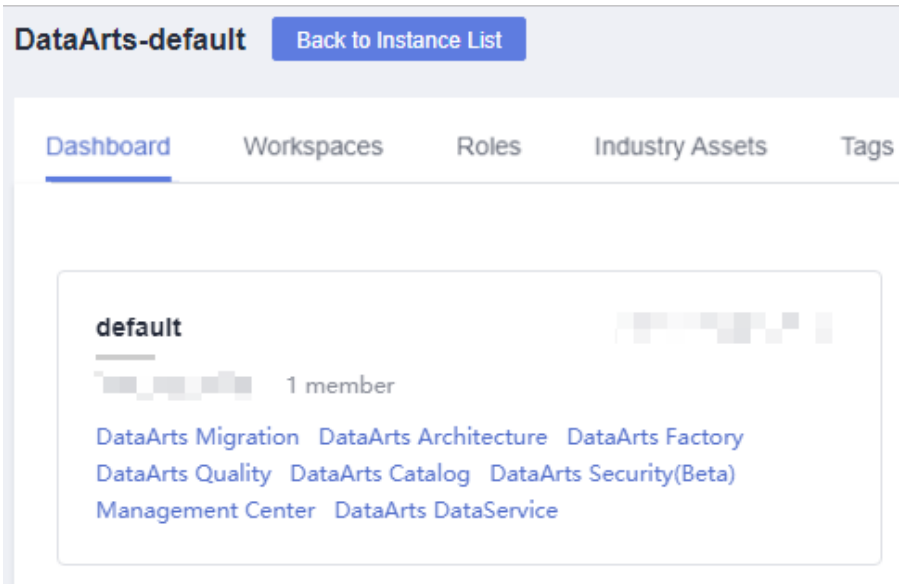
   NOTE

   The API unpublishing time must be later than its minimum retention period. Authorized users will be notified of the unpublishing. If all authorized users process the notifications in the review center or unbind the API from their apps, the API will be directly unpublished. Otherwise, the API will be forcibly unpublished when the minimum retention period ends.

Deleting APIs

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-42 DataArts DataService



2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **API Catalogs**. On the page displayed, select the API you want to delete and click **Delete**.

   NOTE

   - Only APIs in an unpublished state can be deleted. APIs in suspended or published state cannot be deleted.
   - A maximum of 1,000 APIs can be deleted at a time.
4. Click **OK** to delete the API.

10.3.5.4 Copying an API

Scenario

You can copy an API to obtain another API with the same configuration.

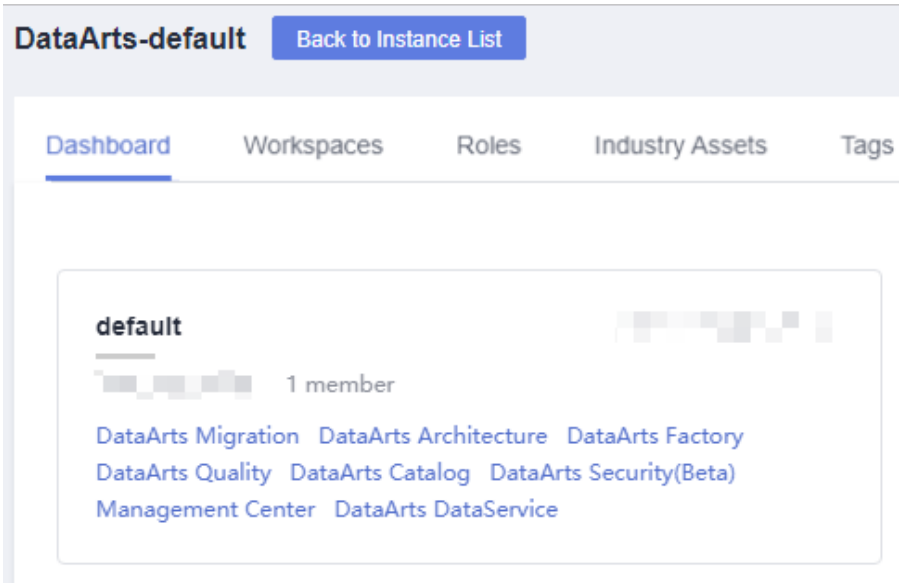
Prerequisites

An API has been created.

Procedure

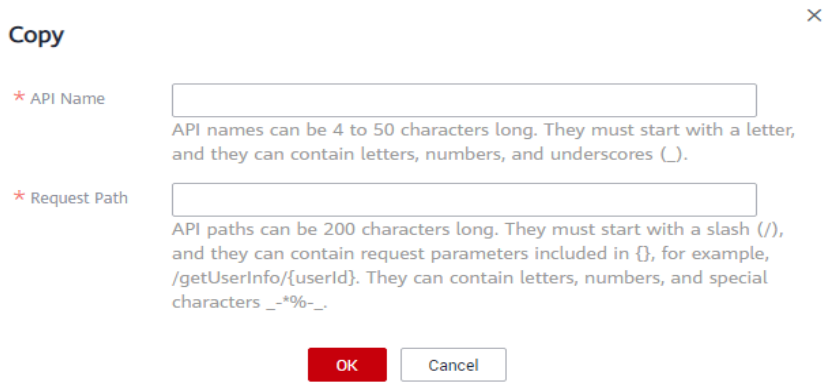
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-43 DataArts DataService



2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **APIs**.
4. Select the target API, click **More** above the API list, and select **Copy**.
5. In the displayed dialog box, enter the new API name and request path, and click **OK**.

Figure 10-44 Copying an API



10.3.5.5 Synchronizing APIs

Operation Scenario

You can synchronize APIs from DataArts DataService Exclusive to Data Map.

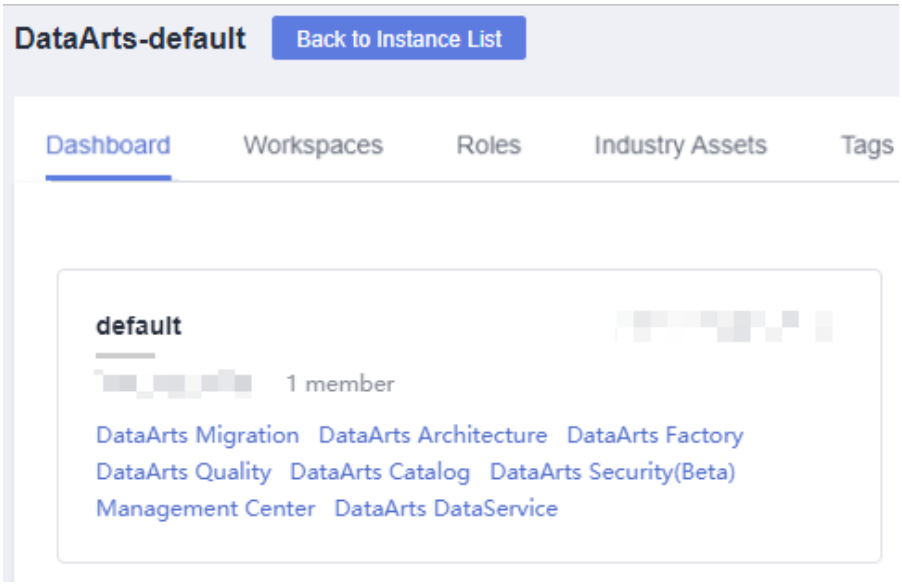
Prerequisites

An API has been created.

Synchronizing APIs to Data Map

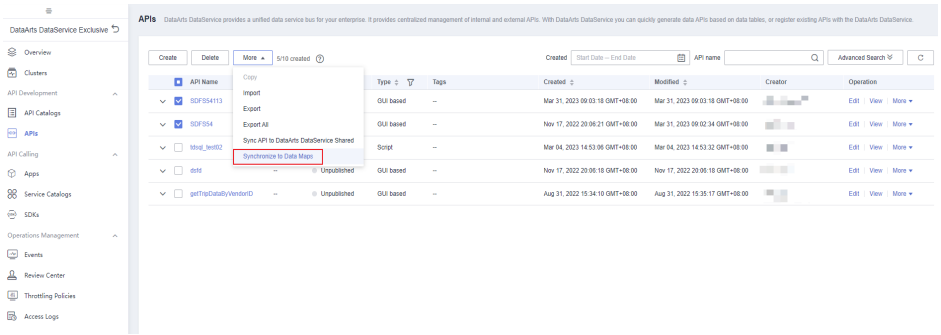
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-45 DataArts DataService



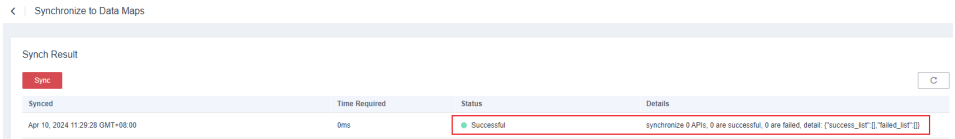
2. In the left navigation pane, choose an edition, for example **DataArts DataService Exclusive**, to access the **Overview** page.
3. Choose **API Development** > **APIs**.
4. Select the target APIs, click **More** above the API list, and select **Synchronize to Data Maps**.

Figure 10-46 Synchronize to Data Maps



5. On the **Synch Result** page, check the API synchronization status and details.

Figure 10-47 Synchronization result



NOTE

- Only published APIs can be synchronized to Data Map.
- Only APIs of the following data sources can be synchronized: DLI, DWS, HBase, and ClickHouse.

10.3.5.6 Exporting All/Exporting/Importing APIs

Operation Scenario

DataArts DataService allows you to import and export (including exporting all) APIs to quickly copy or migrate existing APIs.

Constraints

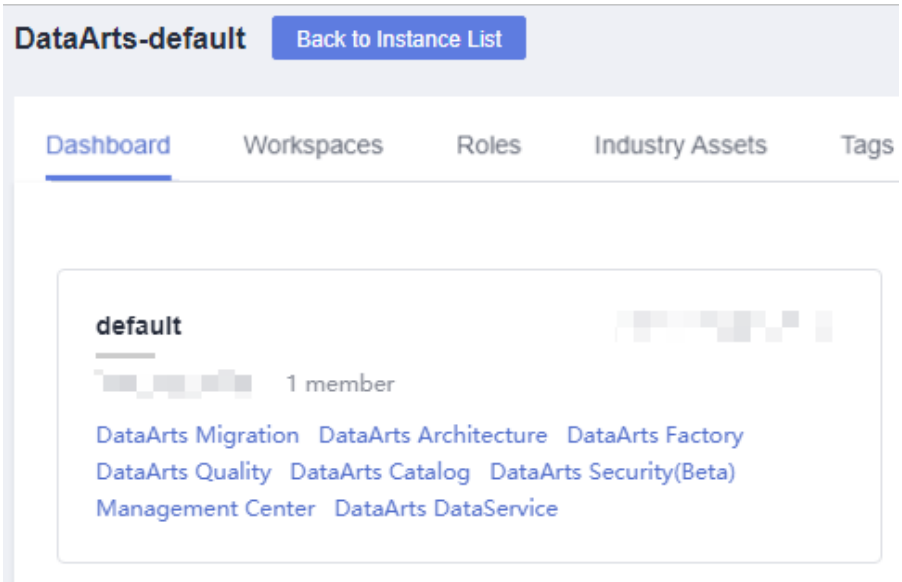
- To export all APIs, you must have the permissions of the **DAYU Administrator** or **Tenant Administrator**.
- All the APIs of a workspace can be exported only once, and only one such export task can be executed within a minute.

Exporting All APIs

You can export all APIs based on the current filter criteria. You must have the permissions of the **DAYU Administrator** or **Tenant Administrator**.

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

   Figure 10-48 DataArts DataService

2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **APIs**.

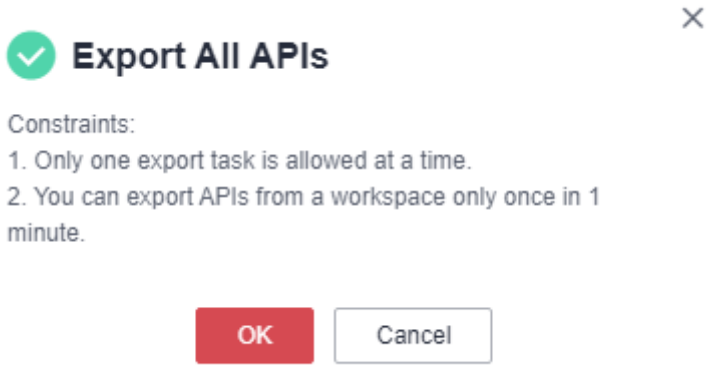
4. Above the API list, choose **More** > **Export All**.

   NOTE

   - To export all APIs, you must have the permissions of the **DAYU Administrator** or **Tenant Administrator**.
   - All the APIs of a workspace can be exported only once, and only one such export task can be executed within a minute.

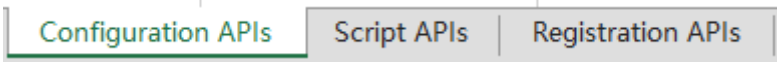
   In the displayed dialog box, click **Yes** to export all the APIs to an Excel file.

   Figure 10-49 Exporting all APIs

5. Open the downloaded Excel file to view the exported APIs. APIs of different types are exported to different sheets.

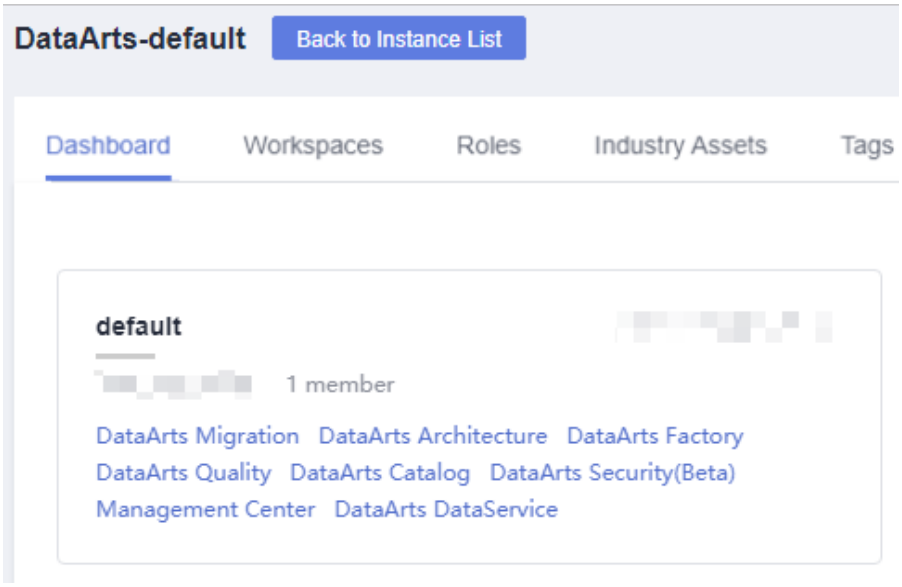
   Figure 10-50 Exported Excel file

Exporting APIs

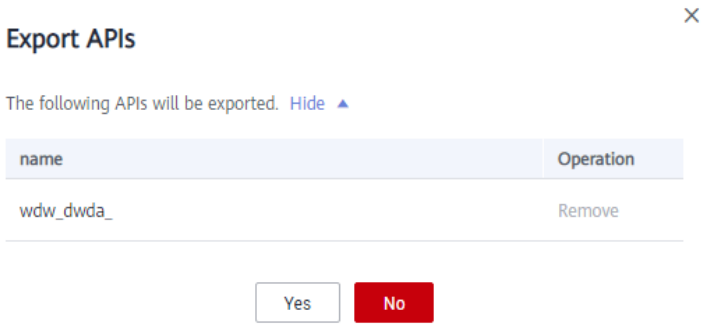
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-51 DataArts DataService



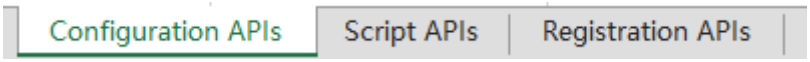
2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **APIs**.
4. Select the target APIs, click **More** above the API list, and select **Export**.
5. In the displayed dialog box, confirm the APIs to export and click **Yes** to export the APIs to an Excel file.

Figure 10-52 Exporting APIs



6. Open the downloaded Excel file to view the exported APIs. APIs of different types are exported to different sheets.

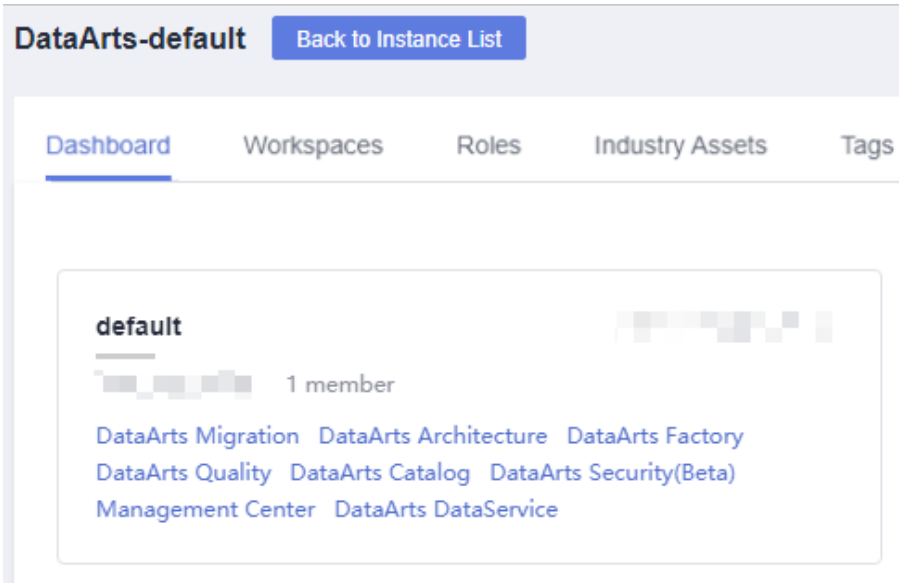
Figure 10-53 Exported Excel file



Importing APIs

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-54 DataArts DataService

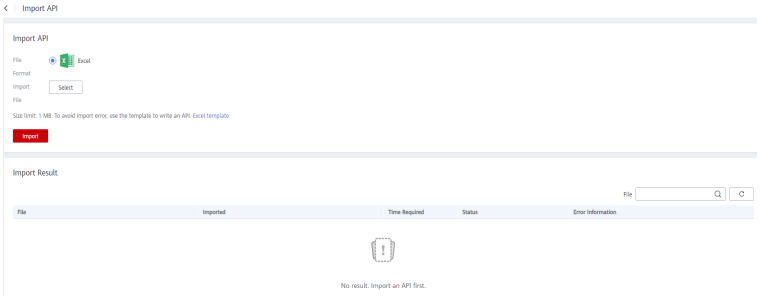


2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **APIs**.
4. Click **More** above the API list and select **Import**.
5. On the displayed page, click **Select**, select an API file, and click **Import**. The import status is displayed in the **Import Result** area.

   NOTE

   The API file can be one exported from another project or an Excel file edited based on the template specifications.

Figure 10-55 Importing APIs



6. After the APIs are imported successfully, you can view them in the API list.

10.3.6 Orchestrating APIs

10.3.6.1 Developing an API Workflow

API orchestration allows you to reorganize and reconstruct APIs in a visualized manner based on specific service logic and processes without compiling code. In this way, you can perform secondary development easily without affecting native

APIs. API orchestration provides you with drag-and-drop and visualized API workflow orchestration capabilities. You can combine multiple APIs into a workflow in serial or parallel mode based on the service logic, invoke the API workflow through the entry API, and obtain the required data.

API orchestration provides more intuitive and efficient design and optimization of business processes, and more convenient secondary development. You can use API orchestration in the following scenarios to simplify development:

- **Map or convert the format of a returned message** through API orchestration.
- **A data request depends on multiple data APIs**: API orchestration reduces the number of API calls, cuts down integration costs, and improves efficiency.

Constraints

- API orchestration is available only for DataArts DataService Exclusive clusters of version 3.0.6 or later.
- Before publishing an API workflow, ensure that all common APIs in the workflow have been published.

Introduction to Operators and Workflows

On the API workflow orchestration page, you can drag various types of operators to the canvas, connect them to orchestrate a workflow based on specific service logic and processes, configure the operators, and save, debug, and publish the workflow.

API orchestration supports five types of drag-and-drop operators: Entry API, Common API, Conditional Branch, Parallel Processing, and Output Processing. A workflow starts with an Entry API operator and ends with an Output Processing operator, with any combination of Common API, Conditional Branch, and Parallel Processing operators in the middle. A workflow must meet the following requirements:

- It starts with and contains only one Entry API operator which can have only one downstream branch.
- It contains at least one Common API operator at the middle layer. The Common API operator has upstream and downstream operators, but can have only one downstream branch.
- Conditional Branch operators are optional and located at the middle layer. They must have at least two branches and can have a maximum of 20 branches. If multiple branches meet a condition, only the first branch is executed.

  The Output Processing operator cannot be the direct downstream operator of a Conditional Branch operator. Instead, Conditional Branch operators obtain the request parameters or result sets of their upstream operators for condition judgment.
- Parallel Processing operators are optional and located at the middle layer. They must have at least two branches and can have a maximum of 20 branches. Failure policies must be configured for Parallel Processing operators.

  The Output Processing operator cannot be the direct downstream operator of a Parallel Processing operator. The logic of multiple branches can be executed at the same time without any impact on each other.

- An API workflow ends with and can have only one Output Processing operator. The direct upstream operator of the Output Processing operator must be a Common API operator, and at least one result mapping must be configured.
- An API workflow cannot have a ring structure or isolated operators. A maximum of 20 layers are supported.

Figure 10-56 API workflow orchestration page

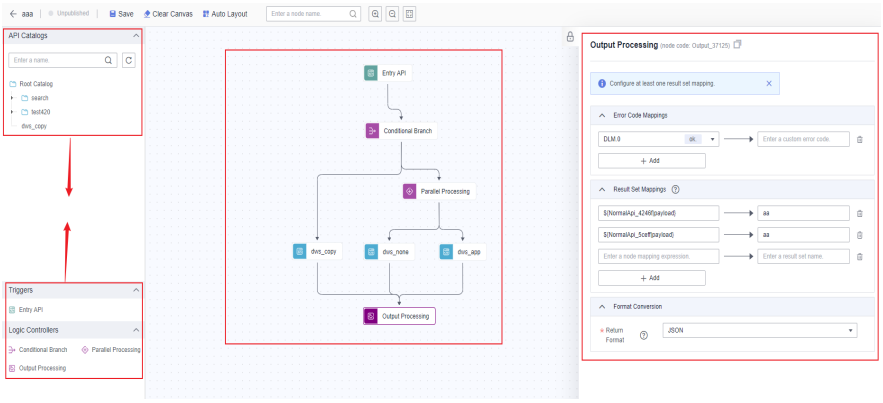


Table 10-12 API workflow operators

Navigation Path	Operator	Mandatory	Description
Triggers	**Entry API**	Yes	An API workflow starts with the Entry API operator. After the API workflow is published, it can be invoked through the Entry API operator. In the Entry API operator, you need to define the API workflow name, URL, parameter protocol, request method, reviewer, security authentication, and request parameters. For details about how to configure an Entry API operator, see **Entry API Operator**.
API Catalogs	**Common API**	Yes	Common API operators are used to perform data query operations. Common APIs are APIs you have created. During API orchestration, you can drag a Common API operator from the API catalog, use the operator to obtain data, and transfer request parameters or result sets as variables. For details about Common API operators, see **Generating an API Using Configuration** or **Generating an API Using a Script or MyBatis**.

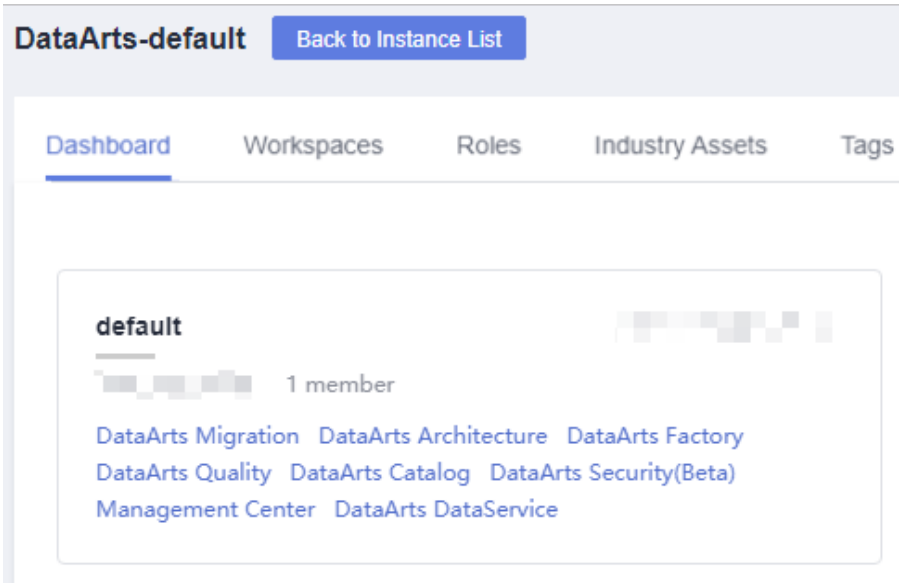
Navigation Path	Operator	Mandatory	Description
Logic Controllers	**Conditional Branch**	No	The Conditional Branch operator obtains the request parameters or result sets of its upstream operator for condition judgment and determines the next branch to be executed based on the defined expression. If the conditions of multiple branches are met, only the first branch is executed. For details about how to configure Conditional Branch operators and expressions, see **Conditional Branch Operator**.
	Parallel Processing	No	The Parallel Processing operator can execute multiple branches at the same time. The branches do not affect each other. For details about how to configure Parallel Processing operators, see **Parallel Processing Operator**.
	Output Processing	Yes	The Output Processing operator maps the error codes and result sets, and converts the format of an API workflow to determine the format of the returned data. For details about how to configure Output Processing operators, see **Output Processing Operator**.

Developing an API Workflow 1: Mapping or Converting the Format of a Returned Message

To convert the result returned by an API from JSON data into an Excel file, perform the following operations:

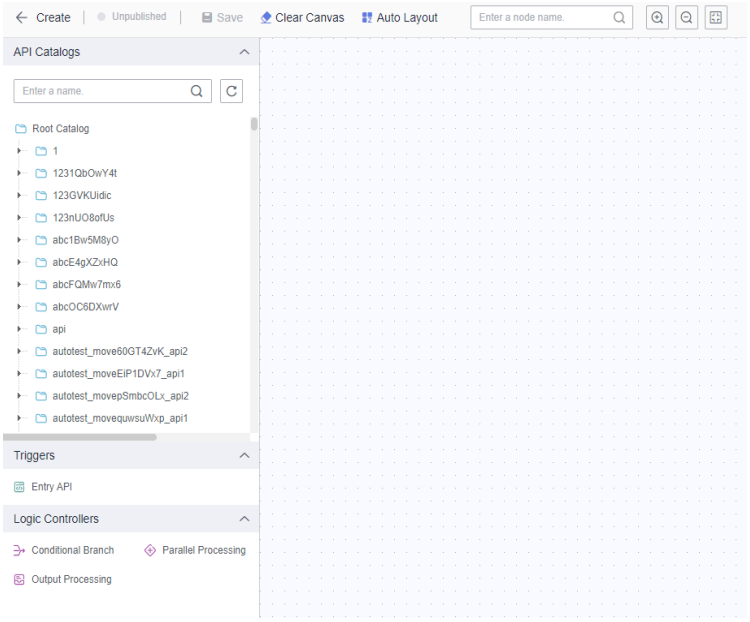
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-57 DataArts DataService



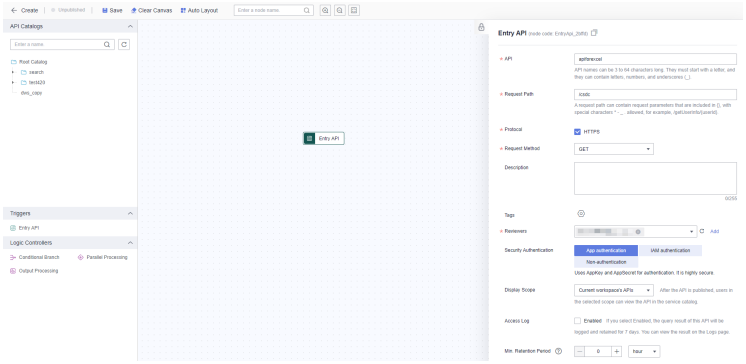
2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **API Orchestration** and click **Create**.

Figure 10-58 API orchestration page



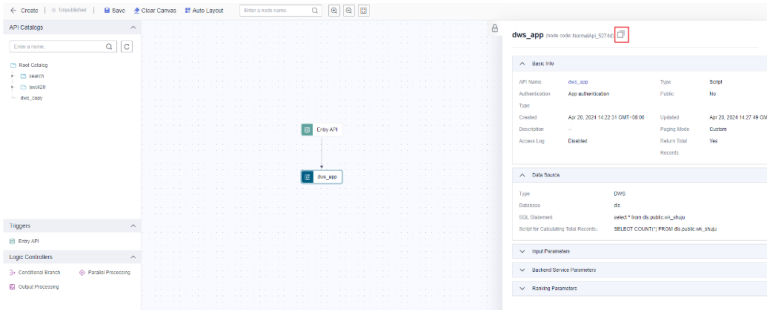
4. Drag the Entry API operator to the canvas, click the operator, and configure its parameters.

Figure 10-59 Configuring the Entry API operator



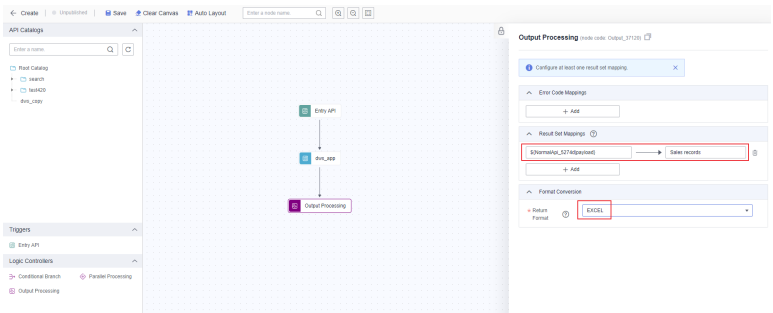
5. Drag the target Common API operator to the canvas and mount it to the Entry API operator. Click the Common API operator and copy its code, for example, **NormalApi_5274d**.

Figure 10-60 Copying code



6. Drag the Output Processing operator to the canvas and mount it to the Common API operator. Click the Output Processing operator configure its parameters.
   - Add a result set mapping. Enter the result of the Common API operator for the mapping expression, for example, **${NormalApi_5274d|payload}** and enter the result set name, for example, **Sales records**.
   - Select **EXCEL** for **Return Format**.

Figure 10-61 Configuring the Output Processing operator



7. Save the API workflow, debug it, and publish it to the cluster. After that, the Entry API operator of the API workflow can be invoked to save the data obtained by the common API to an Excel file.

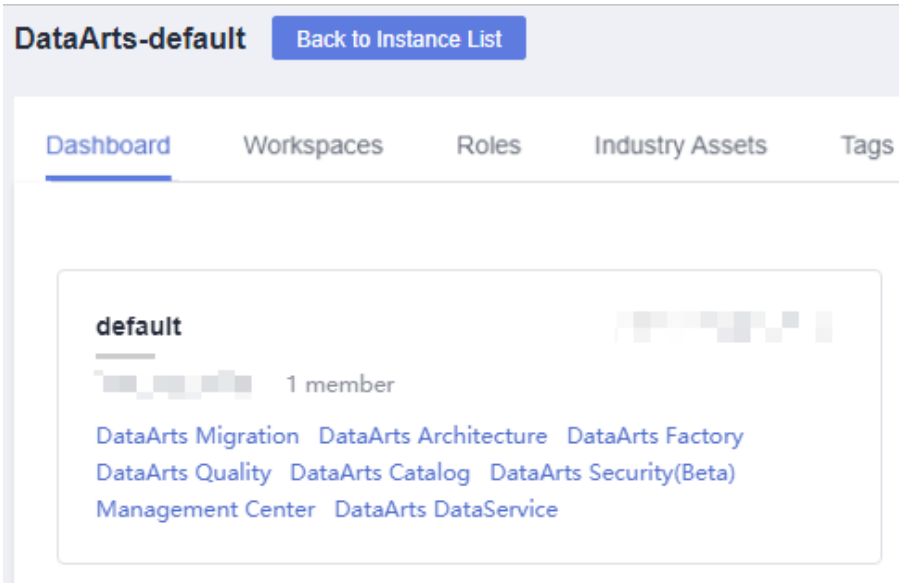
Developing an API Workflow 2: A Data Request Depends on Multiple Data APIs

A department of an e-commerce company wants to provide supplier information and sales rating data for users in area1 and provide retailer information for users in other areas.

The following APIs are available: AreaInformation, SupplierInformation, SalesRating, and RetailerInformation. You can create an API workflow that meets the department's demands by performing the following steps:

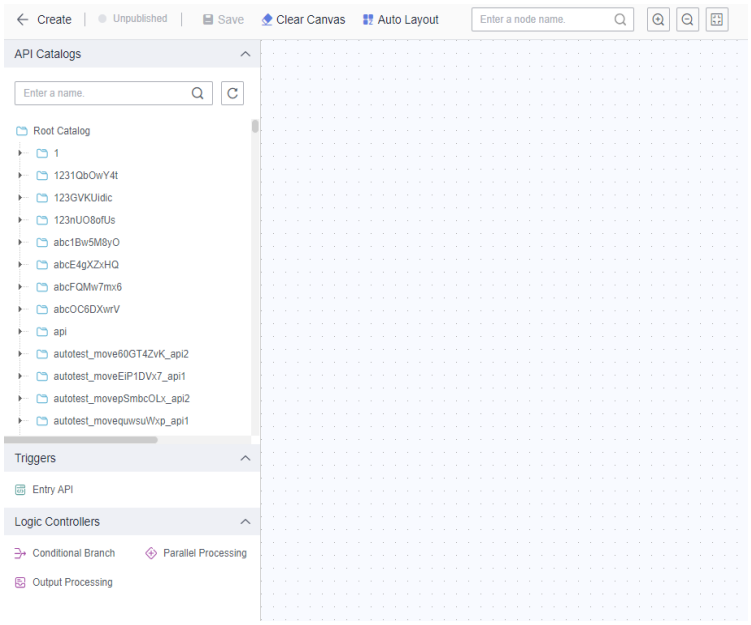
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

   Figure 10-62 DataArts DataService



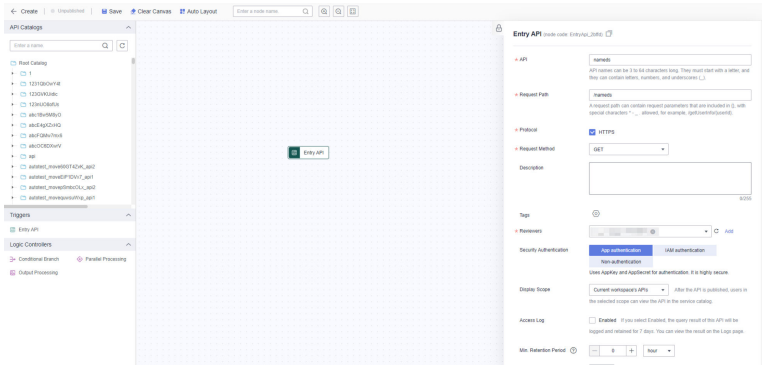
2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Development** > **API Orchestration** and click **Create**.

Figure 10-63 API orchestration page



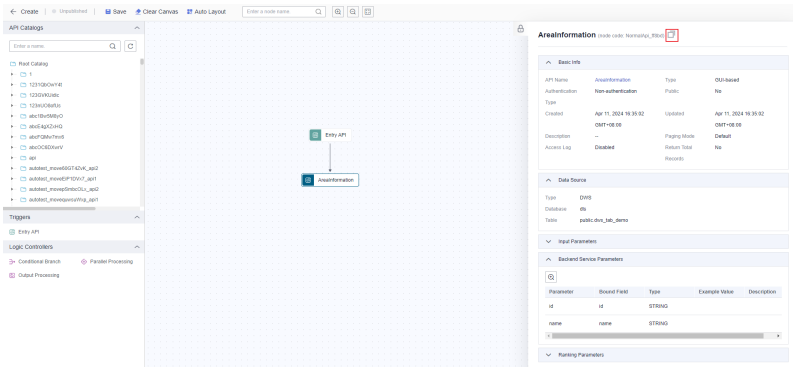
4. Drag the Entry API operator to the canvas, click the operator, and configure its parameters.

Figure 10-64 Configuring the Entry API operator



5. Drag the AreaInformation API operator in the API catalogs to the canvas and mount it to the Entry API operator. Click the AreaInformation API operator and copy its code, for example, **NormalApi_ff8bd**.

Figure 10-65 Copying the code of the AreaInformation operator

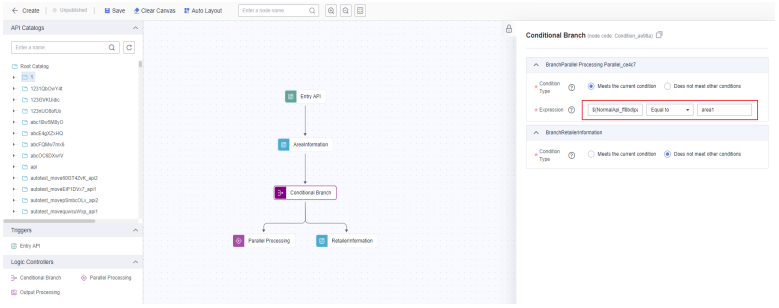


6. Drag the Conditional Branch operator to the canvas, mount it to the AreaInformation operator, and mount the Parallel Processing operator and RetailerInformation operator to the Conditional Branch. The code of the RetailerInformation operator is **NormalApi_de62d**.

   Click the Conditional Branch operator on the canvas and configure its parameters.

   - For the Parallel Processing operator, set **Condition Type** to **Meets the current condition** and **Expression** to **${NormalApi_ff8bd|payload.data[0].area}**. The expression is used to obtain the field value in the first row and the **area** column in the result set of the AreaInformation API. If the obtained field value is **area1**, the Parallel Processing operator is executed.
   - For the RetailerInformation operator, set **Condition Type** to **Does not meet other conditions**. If the conditions of the Parallel Processing operator are not met, the RetailerInformation operator is executed.

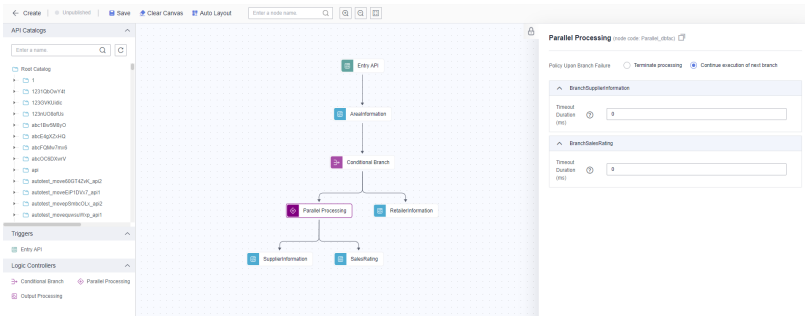
Figure 10-66 Configuring the Conditional Branch operator



7. Drag the SupplierInformation API and SalesRating API operators in the API catalogs to the canvas, and mount them to the Parallel Processing operator. The code of the SupplierInformation operator is **NormalApi_3ad5c** and that of the SalesRating operator is **NormalApi_01e7e**.

   Click the Parallel Processing operator and set **Policy Upon Branch Failure** and **Timeout Duration** for the SupplierInformation and SalesRating operators. (retain their default values.) When the Parallel Processing operator is executed, the SupplierInformation and SalesRating operators are both scheduled.

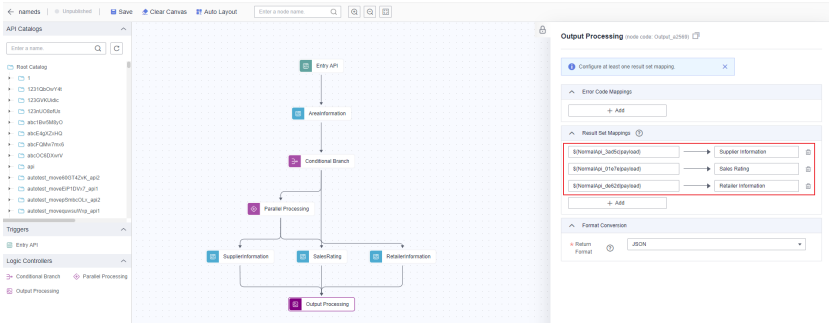
Figure 10-67 Configuring the Parallel Processing operator



8. Drag the Output Processing operator to the canvas and mount it to the three Common API operators. Click the Output Processing operator and add result set mappings.

   Add three mappings to output the results of the three Common API operators. Set the expressions of the mappings to the results of the corresponding Common API operators, for example, **${NormalApi_3ad5c|payload}**, **${NormalApi_01e7e|payload}**, and **${NormalApi_de62d|payload}**, and set the result set names.

Figure 10-68 Configuring the Output Processing operator



9. Save the API workflow, debug it, and publish it to the cluster. After that, the Entry API operator of the API workflow can be invoked to return different information for users in different areas.

Related Operations

- Editing an API workflow: On the API workflow list page, locate a workflow, and click **Edit** in the **Operation** column. On the displayed page, orchestrate the workflow again or modify it.
- Viewing API workflow authorization: On the API workflow list page, locate a workflow and click **View** in the **Operation** column to access the API information page, where you can authorize the workflow.

  If the app authentication mode is used for the Entry API, you must **create an app** and **authorize the app to use the API** before invoking the API workflow. The workflow authorization method is basically the same as the API authorization method. For details, see **authorizing an app to use an API**.
- Debugging an API workflow: On the API workflow list page, locate a workflow, click **More** in the **Operation** column, and select **Debug**.

Add request parameters and click **Test**. The response of the API call is displayed in the result output area on the right. The workflow debugging process is basically the same as the API debugging process. For details, see **Debugging an API**.

- Publishing an API workflow: On the API workflow list page, locate a workflow, click **More** in the **Operation** column, and select **Publish**.

  An API workflow is available only after it is published. The workflow publishing process is basically the same as the API publishing process. For details, see **Publishing an API**.

- Unpublishing/Deleting an API workflow: On the API workflow list page, locate a workflow, click **More** in the **Operation** column, and select **Unpublish** to unpublish the workflow. Select a workflow and click **Delete** above the workflow list to delete the workflow.

  If you want to stop a published API workflow from providing services, you can unpublish it. The authorization information will not be retained after the API workflow is unpublished. If you no longer need the suspended API, you can delete it. The deletion cannot be undone. The process of unpublishing/deleting an API workflow is basically the same as that of unpublishing/deleting an API. For details, see **Unpublishing/Deleting APIs**.

- Suspending/Restoring an API workflow: On the API workflow list page, locate a workflow, click **More** in the **Operation** column, and select **Suspend** or **Restore**.

  To edit or debug a published API workflow, you must suspend the API workflow first. After the API workflow is suspended, its authorization information is retained. You can still edit and debug the API workflow. You can resume the API workflow so that it can continue to provide services. The process of suspending/resuming an API workflow is basically the same as that of suspending/resuming an API. For details, see **Suspending/Restoring an API**.

- Displaying an API workflow: On the API workflow list page, locate a workflow, click **More** in the **Operation** column, and select **Display**.

  Then you can set the visibility scope of the API workflow in the service catalog. The process of setting the visibility scope of an API workflow is basically the same as that of setting the visibility scope of an API. For details, see **Displaying an API**.

- Copying an API workflow: On the API workflow list page, locate a workflow, click **More** above the list, and select **Copy**.

  By copying an API workflow, you can obtain an API workflow with the same configuration as the source API workflow. The processing of copying an API workflow is basically the same as that of copying an API. For details, see **Copying an API**.

- Synchronizing an API workflow to Data Map: On the API workflow list page, locate a workflow, click **More** above the list, and select **Synchronize to Data Map**.

  This allows you to view API workflows in Data Map. The processing of synchronizing an API workflow to Data Map is basically the same as that of synchronizing an API to Data Map. For details, see **Synchronizing APIs to Data Map**.

10.3.6.2 Entry API Operator

An API workflow starts with the Entry API operator. After the API workflow is published, it can be invoked through the Entry API operator. In the Entry API operator, you need to define the API workflow name, URL, parameter protocol, request method, reviewer, security authentication, and request parameters.

Table 10-13 Entry API operator parameters

Parameter	Descriptions
API	Entry API name, that is, API workflow name An API name consists of 3 to 64 characters and starts with a letter. Only letters, numbers, and underscores (_) are allowed.
Request Path	Entry API access path, that is, API workflow access path, for example, **/getUserInfo** It is the part between the domain name and query parameters in the URL of a request path, for example, **/blogs/*xxxx*** shown in the following figure. **Figure 10-69** API access path in the URL  Braces ({}) can be used to identify parameters in a request path as wildcard characters. For example, **/blogs/{blog_id}** indicates that any parameter can follow **/blogs**. **/blogs/188138** and **/blogs/0** can both match **/blogs/{blog_id}**, and are processed by this API. In addition, duplicate request paths are not allowed for the same domain name. When a path parameter is used as a wildcard, the name is not unique. For example, **/blogs/{blog_id}** and **/blogs/{xxxx}** are considered as the same path.
Protocol	Protocol used to transmit requests. The exclusive edition supports HTTPS. HTTPS is recommended. It is an HTTP-based protocol with SSL or TLS encryption verification. It can effectively verify identities and protect data integrity. To access HTTPS APIs, you need to configure related SSL certificates or skip SSL verification.

Parameter	Descriptions
Request Method	HTTP request method, indicating the type of the requested operation, such as GET and POST. The method complies with the resultful style. • **GET** requests the server to return specified resources. This method is recommended. • **POST** requests the server to add resources or perform special operations. This method is used only for API registration. The POST request does not have a body. Instead, it involves transparent transmission.
Description	A brief description of the API to create.
Tags	API tag. The tag is used to mark the API attributes. After the API is created, you can quickly search for the API by tag. A maximum of 20 tags can be set for an API.
Reviewers	A reviewer who has permissions to review APIs. Click **Add** to enter the **Review Center** page and click **Add** on the **Reviewers** tab page to add a reviewer.
Security Authentication	When creating an API, you can select one of the following authentication modes. The three modes differ in how the API is called. You are advised to use **App Authentication**, which is more secure that the other two modes. • **App authentication**: App authentication is used for calling an API. The AppKey & AppSecret is used for authentication. It is highly secure. When **App authentication** is used, an SDK is required for access. Java, Go, Python, JavaScript, C#, PHP, C++, C, and Android SDKs are available. For details about how to call APIs in each language, see **Calling APIs Through App Authentication**. • **IAM authentication**: IAM authenticates API requests. This mode is available only for Huawei cloud users. The security level is medium. When using IAM authentication, you need to call the **Obtaining a User Token** API of IAM to obtain a token, add the **X-Auth-Token** parameter with the obtained token as the value to the request header, and use an API calling tool or SDK to call released APIs. • **Non-authentication**: No authentication is required. This mode allows all users to access APIs, which may pose security risks. It is recommended only for testing APIs. If the caller is not a trusted user, there is a risk of data leakage, breakdowns caused by high concurrent access, SQL injection, and others. This mode does not require any authentication information. You can use an API calling tool or SDK to directly call an API by specifying required parameters.

Parameter	Descriptions
Display Scope	After the API is published, all users in the selected scope can view the API in the service catalog. ● Current workspace APIs ● Current project APIs ● Current tenant's APIs
Access Log	If you select this option, the API query result will be recorded and retained for seven days. You can choose **Operations Management** > **Access Logs** and select the request date to view the logs.
Min. Retention Period	Minimum retention period of the API publishing status, in hours. Value **0** indicates that the retention period is not limited. You can suspend, unpublish, or cancel authorization for an API only after the minimum retention period ends. The system notifies the authorized users. If all authorized users have processed the notifications or unbound the API from their apps, the API will be suspended or unpublished, or the API authorization will be canceled. Otherwise, the system will forcibly suspend, unpublish, or cancel authorization for the API when the minimum retention periods ends. For example, if the minimum retention period is set to 24 hours, the API can be suspended 24 hours after it is published. If the authorized user handles the notifications in the review center or unbind the API from the app, the API will be directly suspended. Otherwise, the API will be forcibly suspended when the minimum retention period ends.

Parameter	Descriptions
Input Parameters	Parameters required for invoking the API workflow. An input parameter consists of the parameter location, parameter type, whether the parameter is mandatory, and the default value. ● The parameter location can be **Query**, **Header**, **Path**, or **Body**. In addition, static parameters are supported. – **Query** is the query parameter following the URL. It starts with a question mark (?) and connects multiple parameters with **&**. – **Header** is located in the request header and is used to transfer current information, for example, host and token. – **Path** is a request parameter in the request path. If you configure a path parameter, you must also add this parameter to the request path. – **Body** is a parameter in the request body and is generally in JSON format. – **Static** is a static parameter that does not change with the value passed by API callers. The parameter value is determined upon API authorization. If the parameter value is not set during authorization, the default value of the API input parameter is used. ● The parameter type can be **Number** or **String**. **Number** corresponds to numeric data types such as int, double, and long. **String** corresponds to text data types such as char, vachar, and text. ● **Mandatory** and **Default Value**: If you select **Yes** for **Mandatory**, parameters must be passed for accessing the API. Otherwise, the default value of the parameter will be used if the parameter is not passed for accessing the API. **NOTE** When defining an input parameter, ensure that the following size requirements are met: ● **Query** and **Path**: 32 KB. ● **HEADER**: The maximum size is 128 KB. ● **BODY**: The maximum size is 128 KB. You need to set input parameters based on the designed request parameters for the API workflow. For example, the request path of the API workflow used to query user information in multiple tables by user ID is **/getUserInfo**. You can configure input parameters as follows: ● If the request parameter for calling the API is **id**, and the information about the user with **id** needs to be returned through the API workflow , configure an input parameter as follows:

Parameter	Descriptions
	1. Click **Add** and enter **id** for **Name**. 2. Set **Parameter Location** to **Query**. 3. Set **Type** to **Number**. 4. Select **Yes** for **Mandatory**. 5. Retain the default value. • If the request parameters for calling the API are **id1** and **id2**, and the user information between **id1** and **id2** needs to be returned through the API workflow, configure input parameters as follows: 1. Click **Add** and enter **id1** for **Name**. 2. Set **Parameter Location** to **Query**. 3. Set **Type** to **Number**. 4. Select **Yes** for **Mandatory**. 5. Retain the default value. 6. Click **Add** again and configure parameter **id2**.

10.3.6.3 Conditional Branch Operator

The Conditional Branch operator obtains the request parameters or result sets of its upstream operator for condition judgment and determines the next branch to be executed based on the defined expression. If the conditions of multiple branches are met, only the first branch is executed.

Table 10-14 Conditional Branch operator parameters

Parameter	Description
Branch 1	
Condition Type	Condition type. • **Meets the current condition**: When data transferred to the conditional branch meets the specified expression, the branch is executed. • **Does not meet other conditions**: When data transferred to the conditional branch does not meet the conditions of any other branch, the branch is executed.
Expression	This parameter is mandatory when **Condition Type** is **Meets the current condition**. The expression consists of the code and variable name of the upstream operator. For details about how to use the expression, see **Defining an Expression**.
Branch 2	

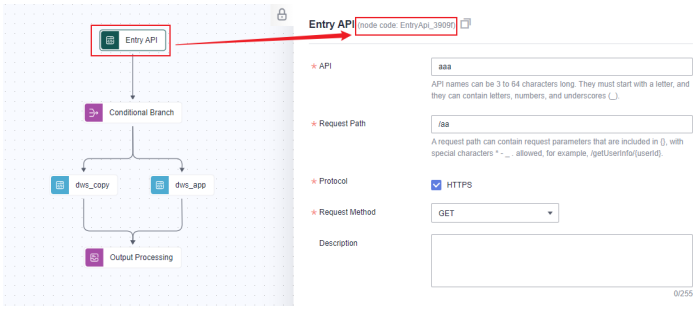
Parameter	Description
Condition Type	Condition type. • **Meets the current condition**: When data transferred to the conditional branch meets the specified expression, the branch is executed. • **Does not meet other conditions**: When data transferred to the conditional branch does not meet the conditions of any other branch, the branch is executed.
Expression	This parameter is mandatory when **Condition Type** is **Meets the current condition**. The expression consists of the code and variable name of the upstream operator. For details about how to use the expression, see **Defining an Expression**.
...	
Branch *n*	
Condition Type	Condition type. • **Meets the current condition**: When data transferred to the conditional branch meets the specified expression, the branch is executed. • **Does not meet other conditions**: When data transferred to the conditional branch does not meet the conditions of any other branch, the branch is executed. • **Meets the current condition**: When data transferred to the conditional branch meets the specified expression, the branch is executed. • **Does not meet other conditions**: When data transferred to the conditional branch does not meet the conditions of any other branch, the branch is executed.
Expression	This parameter is mandatory when **Condition Type** is **Meets the current condition**. The expression consists of the code and variable name of the upstream operator. For details about how to use the expression, see **Defining an Expression**.

Defining an Expression

When defining the expression of a conditional branch, you need to configure a variable expression. Variable expressions are available for Entry API and Common API operators, but unavailable for Conditional Branch, Parallel Processing, and Output Processing operators. The standard expression format is ${***Node code*|*Variable name***}. For details about how to define an expression, see **Table 10-15**.

- **Node code**: It is dynamically allocated by the system and cannot be changed. You can click a node in the API orchestration canvas to view the node code and click to copy the node code.

  Figure 10-70 Viewing the node code



- **Variable name**: Supported variables include request parameter values and result set parameters. For details, see **Table 10-15**.

Table 10-15 Methods for defining an condition expression

Operator	Variable Expression	Example Value
Entry API	Obtain the value of the request parameter of the entry API: ${***Node code*|*Input parameter name***}. **NOTE** This expression is supported for POST requests whose input parameters are located in Query, Header, Path, or Body.	If the node code of the entry API is **EntryApi_3909f**, and the input parameter **userId** is located in Path, set the expression for obtaining the value of the request parameter to **${EntryApi_3909f|userId}**.

Operator	Variable Expression	Example Value
Common API	1. Obtain the value of the request parameter of the common API: ${***Node code*|*Input parameter name***}. **NOTE** This expression is supported for POST requests whose input parameters are located in Query, Header, Path, or Body. 2. Obtain the result sets and related variables of common APIs: ● ${***Node code***|payload.success}: checks whether the query status of a common API is successful. The result is true or false. ● ${***Node code***|payload.rowSize}: obtains the number of rows in the query result set of a common API. ● ${***Node code***|payload.columnSize}: obtains the number of columns in the query result set of a common API. ● ${***Node code***|payload.columnNames}: obtains the column names in the query result set of a common API. ● ${***Node code***|payload.data[***n-1***].***id***}: obtains the value of row n in the *id* column in the query result set of a common API.	● If the node code of a common API is **NormalApi_4246f**, and the input parameter **userId** is located in Path, set the expression for obtaining the value of the request parameter to **${NormalApi_4246f\|userId}**. ● If the node code of a common API is **NormalApi_4246f**, and the value is a one-dimensional array of multiple rows and a single column, set the expression for obtaining the values of the first row in the result set to **${NormalApi_4246f|payload.data[0]}**. ● If the node code of a common API is **NormalApi_4246f**, and the value is a two-dimensional array of multiple rows and columns, set the expression for obtaining the value in the first row and **price** column in the result set to **${NormalApi_4246f|payload.data[0].price}**.

For example, if there are three sequential nodes, A (entry API), B (common API), and C (conditional branch), and node C needs to obtain the request parameter values of node A and the output values of node B:

- If the code of node A is **EntryApi_3909f**, and the location of input parameter **userId** is **Path**, set the expression for obtaining the request parameter value of node A as follows:

  ${EntryApi_3909f|userId}

- If the code of node B is **NormalApi_4246f**, and the value is a two-dimensional array of multiple rows and columns, set the expression for obtaining the value in the first row and **name** column in the result set of node B as follows:

  ${NormalApi_4246f|payload.data[0].name}

10.3.6.4 Parallel Processing Operator

The Parallel Processing operator can execute multiple branches at the same time. The branches do not affect each other.

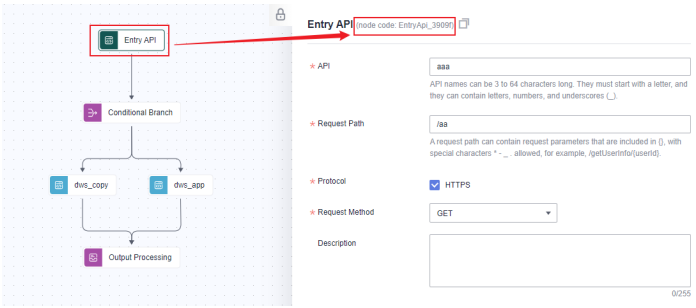
Table 10-16 Parallel Processing operator parameters

Parameter	Description
Policy Upon Branch Failure	Policy for processing the API workflow when one of the parallel branches fails • **Terminate processing**: terminates the API workflow if any branch fails. • **Continue execution of next branch**: continues to execute other branches and subsequent operators even if a branch fails. If all branches fail and no operators can be executed, the API workflow status becomes failed.
Branch 1	
Timeout Duration (ms)	If the execution of the current branch times out, the branch status becomes failed. The default value is 0, indicating that there is no time limit.
Branch 2	
Timeout Duration (ms)	If the execution of the current branch times out, the branch status becomes failed. The default value is 0, indicating that there is no time limit.
...	
Branch *n*	
Timeout Duration (ms)	If the execution of the current branch times out, the branch status becomes failed. The default value is 0, indicating that there is no time limit.

10.3.6.5 Output Processing Operator

The Output Processing operator maps the error codes and result sets, and converts the format of an API workflow to determine the format of the returned data.

Table 10-17 Output Processing operator parameters

Parameter	Mandatory	Description
Error Code Mappings	No	Error codes returned by DataArts DataService can be mapped to custom information, for example, error code **DLM.0** can be mapped to **OK**.
Result Set Mappings	Yes	The result set names of one or more Common API operators can be mapped to custom names which will be used in the JSON string or file name. Result sets that are not mapped will not be output to the final returned result. The node mapping expression is in **${*Node code*\|payload}** format. You can obtain the node code by clicking a node in the API orchestration canvas, and copy the code by clicking  . **Figure 10-71** Viewing the node code  For example, if the node code is **NormalApi_5a256**, set the node mapping expression to **${NormalApi_5a256|payload}** and the result set name to **sales record**.
Format Conversion	No	By default, a workflow result is a JSON string. You can export each mapped result set to a CSV, TXT, Excel, or XML file, or export all mapped result sets to a .zip file. Resumable data transfer is not supported during export.

10.3.7 Creating Throttling Policies

Scenario

A throttling policy limits the maximum number of times that an API can be called within a specific period. Throttling policies can protect the backend service from getting overloaded. Currently, API throttling can limit the number of API calls by user, application, and time period.

To ensure the stability of services, you can create throttling policies to control the calls made to specified APIs. Throttling policies take effect for an API only if they are bound to the API.

NOTE

An API can be bound to only one throttling policy in an environment, but each throttling policy can be bound to multiple APIs.

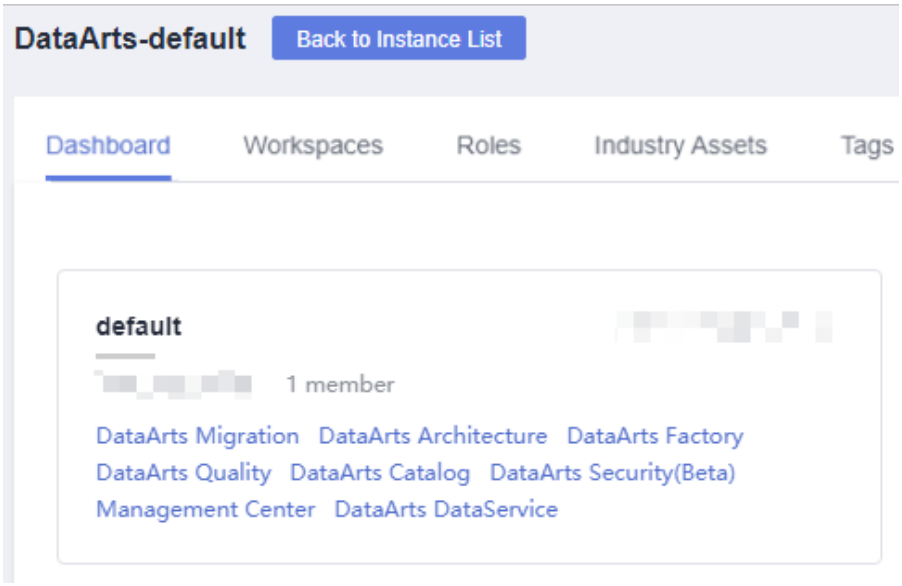
Prerequisites

The API to be bound has been published.

Creating a Throttling Policy

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

   Figure 10-72 DataArts DataService

2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. On the displayed page, choose **Throttling Policies** from the left navigation bar.
4. On the displayed page, click **Create**. Set the parameters listed in **Table 10-18**.

Figure 10-73 Creating a throttling policy

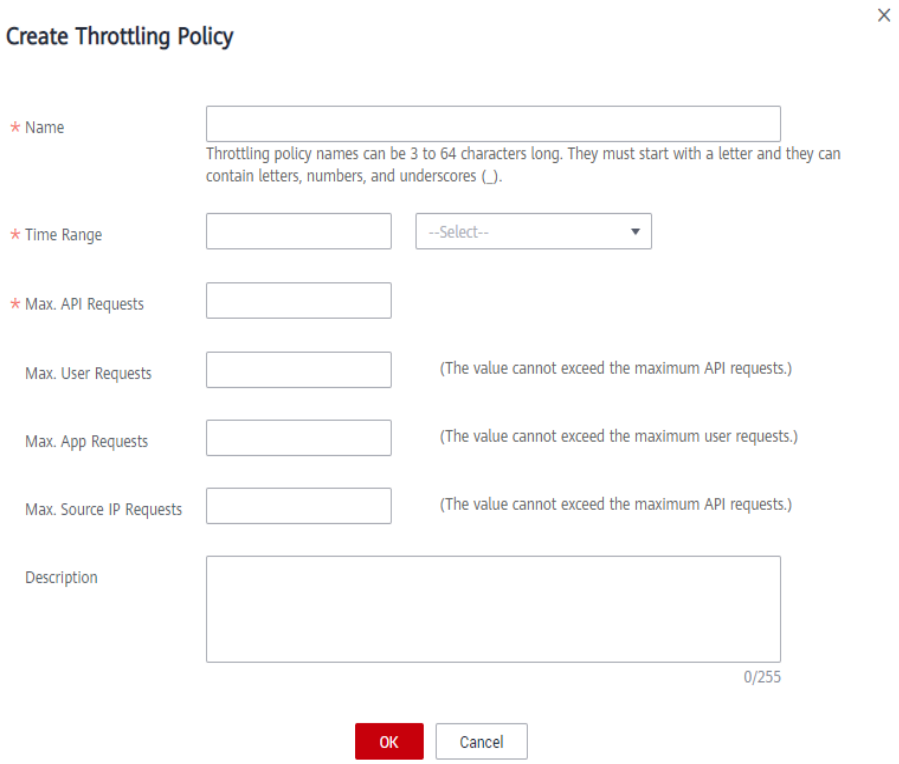


Table 10-18 Parameters

Parameter	Description
Name	The throttling policy name.
Time Range	The time duration for limiting the number of API calls • Used together with **Max. API Requests** to specify the total number of times an API can be called within a time period. • Used together with **Max. User Requests** to specify the number of times an API can be called by a user within a time period. • Used together with **Max. App Requests** to specify the total number of times an API can be called by an app within a time period.
Max. API Requests	The maximum number of times an API can be called within the specified time period. Used together with **Time Range** to specify the maximum number of times an API can be called within the period.

Parameter	Description
Max. User Requests	The maximum number of times an API can be called by a user within the specified period. • The value of this parameter must be less than that of **Max. API Requests**. • Used together with **Time Range** to specify the maximum number of times an API can be called by a user within the specified period.
Max. App Requests	The maximum number of times an application can be called by a user within the specified period. • The value of this parameter must be less than that of **Max. User Requests**. • Used together with **Time Range** to specify the maximum number of requests an app can make within the specified period.
Description	A description of the throttling policy to be created

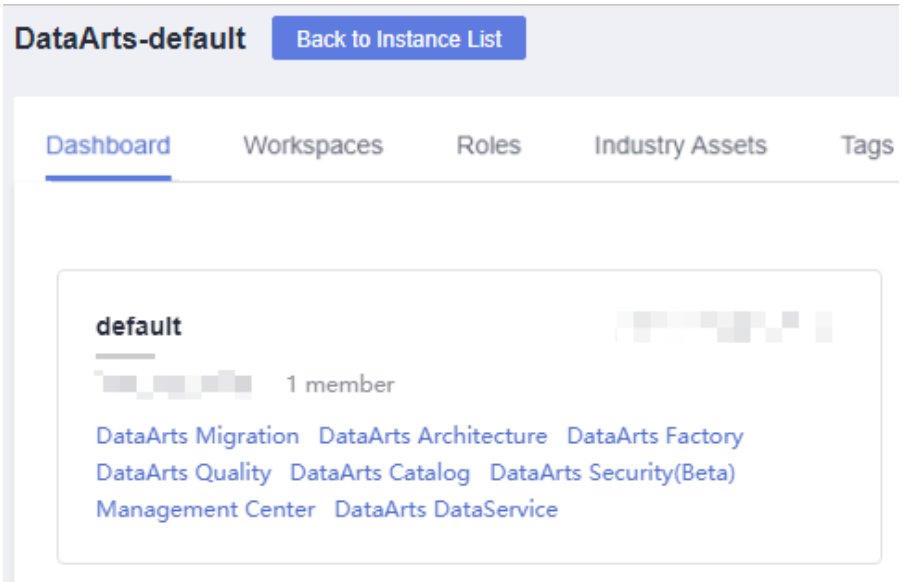
5. Click **OK**.

   After the throttling policy is created, it is listed in the throttling policy list. Bind the throttling policy to an API to limit the access traffic.

Binding a Throttling Policy to an API

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

   Figure 10-74 DataArts DataService



2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.

3. Choose **Throttling Policies** from the left navigation bar.
4. Bind a throttling policy to an API in either of the following ways:
   - Locate the throttling policy to be bound and click **Associate with APIs**.
   - Click the target policy name to go to its details page and click **Associate with APIs** on the **List of Associated APIs** tab page.
5. Enter an API group and API name to search for the target API.
6. Select the API and click **OK**.

   NOTE

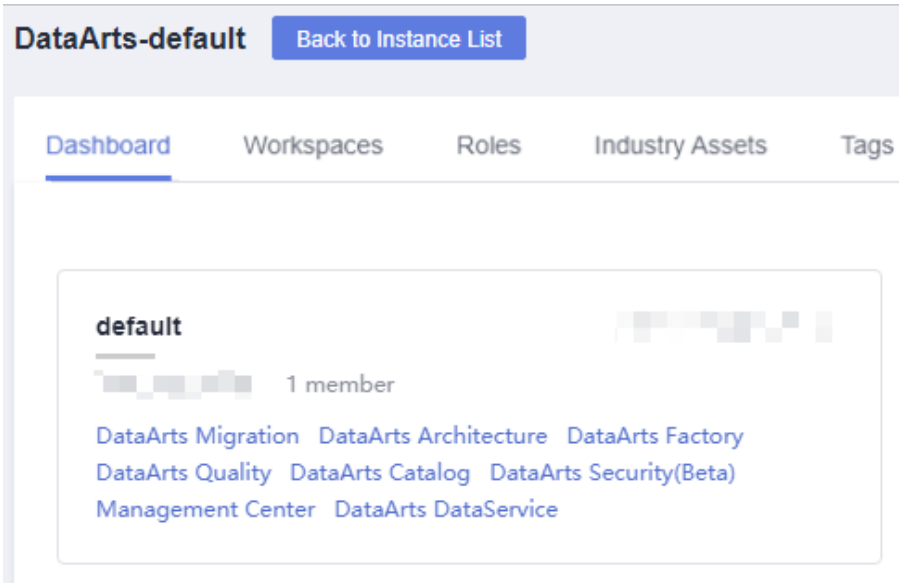
   If a throttling policy is no longer needed, click **Unbind** on the **List of Associated APIs** tab page. To unbind multiple APIs at a time, select the APIs to be unbound and click **Unbind**. Up to 1000 APIs can be unbound at a time.

Deleting a Throttling Policy

You can delete a throttling policy if it is no longer needed.

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

   Figure 10-75 DataArts DataService



2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. On the displayed page, choose **Throttling Policies** from the left navigation bar.
4. On the displayed page, locate the policy you want to unbind and click **Delete** in the **Operation** column.

NOTE

- Throttling policies bound to APIs cannot be deleted. Therefore, you need to unbind them from APIs before deleting them.
- To delete multiple throttling policies at a time, select the policies, and click **Delete**. Up to 1000 throttling policies can be deleted at a time.

5. Click **Yes**.

10.4 Calling APIs

Overview

To call an API, perform the following operations:

1. Obtain an API.

   Obtain the API from the service catalog. An API can be called only after it is published.

2. (Optional) Create an application and get authorized.

   For an API that is accessed using application authentication, you need to **create an application** and **authorize the application to use the API**. When you call an API, DataArts DataService verifies your identity based on the key pair (AppKey and AppSecret) of the created application.

3. **Call the API**.

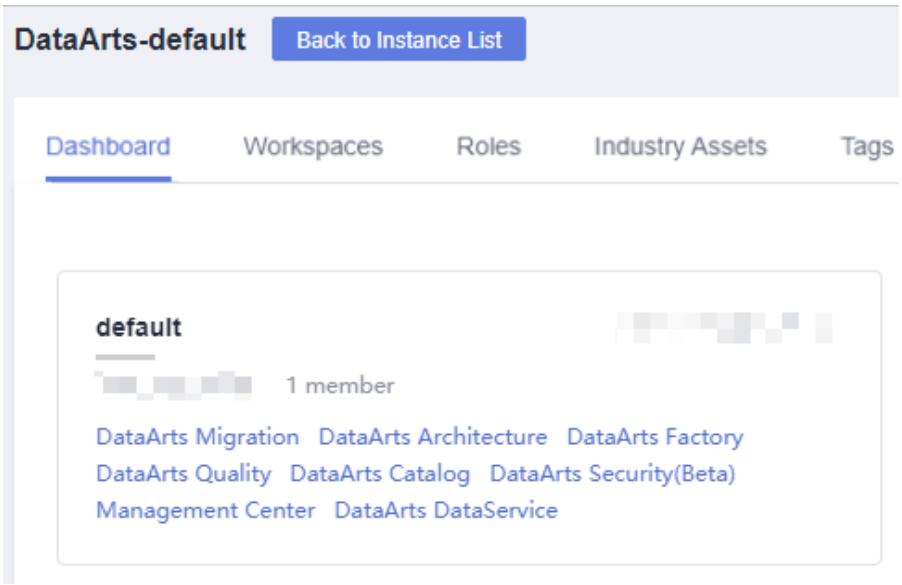
   After completing the preceding steps, you can call the API.

(Optional) Creating an App

Perform this operation when the API to be called uses app authentication.

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-76 DataArts DataService



2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **API Calling** > **Apps**. On the page displayed, click **Create**. The **Create App** dialog box is displayed. Set the parameters listed in **Table 10-19**.

   Table 10-19 App information

Parameter	Description
Name	The name of the application to create.
Type	**IAM**: IAM authentication is used, which means access using a token. **APP**: access through app authentication
Description	A description of the application to create.

4. Click **OK**.

   After the application is created, its name and ID are displayed in the application list.
5. Click an application name, and view the **AppKey** and **AppSecret** on the displayed application details page.

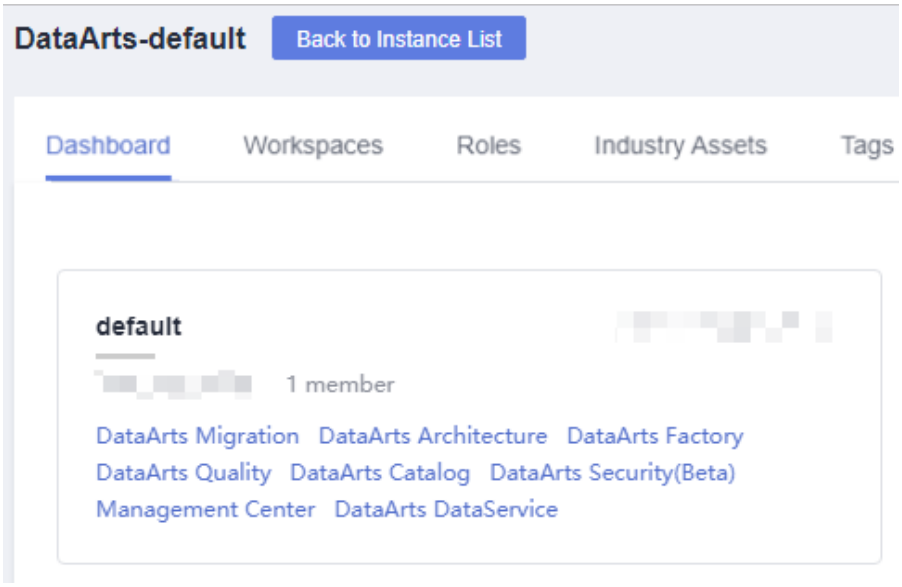
   Figure 10-77 Application details page

(Optional) Authorizing an App to Use an API

Perform this operation when the API to be called uses app authentication.

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-78 DataArts DataService



2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Authorize an application to use an API in either of the following ways:

   Giving API authorization:

   a. Choose **API Development** > **APIs**.
   b. Locate the row that contains the API to be bound, and click **View**.

      On the page displayed, click **Authorize**.
   c. (Optional) If **Parameter Location** was set to **Static** for an input parameter during API creation, you must set a static parameter value. If you do not set a value, the default value of the API input parameter is used.
   d. Set an expiry time, select an application, and click **OK**.

   Applying for authorization:

   a. Choose **API Calling** > **Service Catalogs** to view all the published APIs.
   b. Click the name of the API you want to bind to an application.
   c. On the page displayed, click **Permission Application**.
   d. (Optional) If **Parameter Location** was set to **Static** for an input parameter during API creation, you must set a static parameter value. If you do not set a value, the default value of the API input parameter is used.
   e. Set an expiry time, select an application, and click **OK**.
   f. After the application is submitted, the authorization takes effect only after it is approved in the review center.
4. After the authorization is complete, view the bound APIs on the application details page.

NOTE

- In the API list, if you no longer access an API through the application, click **Unbind** in the **Operation** column.
- To test an API to which the application is bound, choose **More** > **Debug** in the **Operation** column
- To extend the authorization period for the bound API, click **Renew**.

Calling an API

When creating an API, you can select one of the following authentication modes. The three modes differ in how the API is called. You are advised to use **App Authentication**, which is more secure that the other two modes.

- **App authentication**: App authentication is used for calling an API. The AppKey & AppSecret is used for authentication. It is highly secure.

  When **App authentication** is used, an SDK is required for access. Java, Go, Python, JavaScript, C#, PHP, C++, C, and Android SDKs are available. For details about how to call APIs in each language, see **Calling APIs Through App Authentication**.
- **IAM authentication**: IAM authenticates API requests. This mode is available only for Huawei cloud users. The security level is medium.

  When using IAM authentication, you need to call the **Obtaining a User Token** API of IAM to obtain a token, add the **X-Auth-Token** parameter with the obtained token as the value to the request header, and use an API calling tool or SDK to call released APIs.
- **Non-authentication**: No authentication is required. This mode allows all users to access APIs, which may pose security risks. It is recommended only for testing APIs. If the caller is not a trusted user, there is a risk of data leakage, breakdowns caused by high concurrent access, SQL injection, and others.

  This mode does not require any authentication information. You can use an API calling tool or SDK to directly call an API by specifying required parameters.

10.5 Configuring Log Dump and Viewing Logs on LTS

Scenario

You can query logs of DataArts DataService APIs, including the request path, request parameters, and response.

NOTE

Currently, logs are only supported for APIs in DataArts DataService Exclusive.

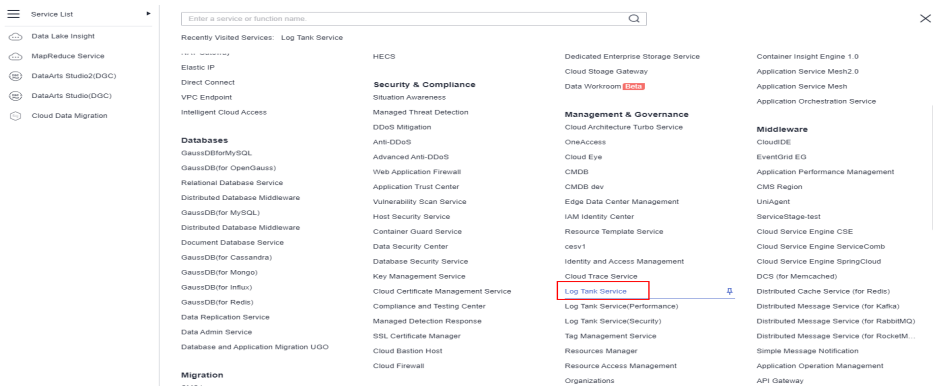
Configuring LTS

To view logs of DataArts DataService APIs, you need to first configure LTS. For details about how to configure LTS, see **Log Tank Service User Guide**.

Step 1 Create a log group on the LTS console.

1. Log in to the management console.
2. Click in the upper left corner and select a region and project.
3. Click **Service List** and click **Log Tank Service** under **Management & Governance**.

   Figure 10-79 Accessing the LTS console



4. In the navigation pane on the left, choose **Log Management**.
5. Click **Create Log Group**. In the displayed dialog box, enter a log group name.
6. Click **OK**.

Step 2 Create a log stream.

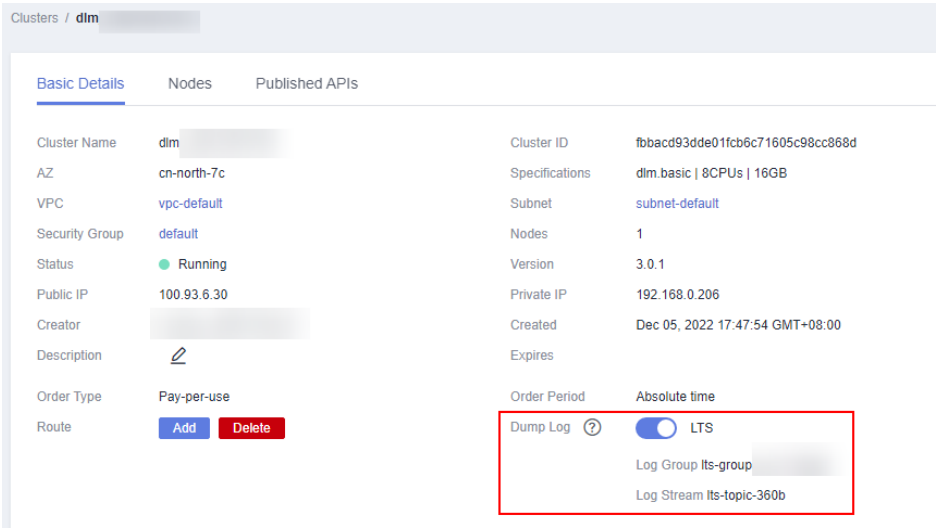
1. Click the name of the created log group.
2. Click **Create Log Stream**. In the displayed dialog box, enter a log stream name.
3. Click **OK**.

----End

Enabling Dump of DataArts DataService Logs

Log in to the DataArts DataService Exclusive console, enter the **Basic Details** page of a cluster, enable **Dump Log**, and select **LTS**.

Figure 10-80 Enabling dump of logs to LTS



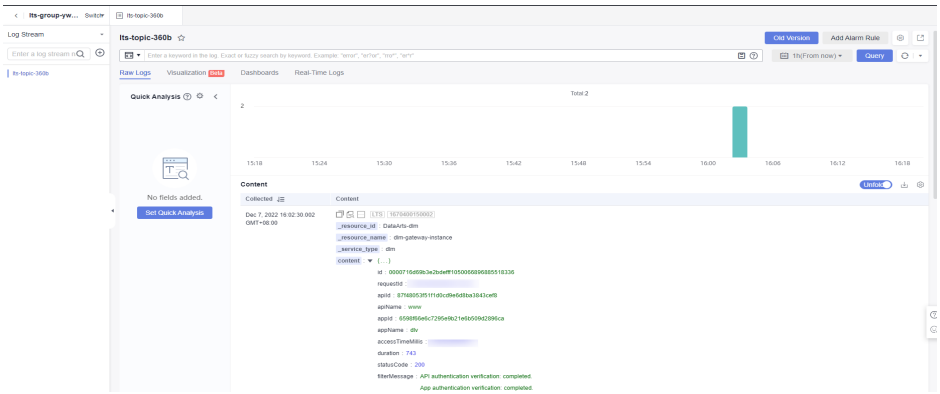
Viewing Access Logs

After configuring log dump, you can view details about access logs.

On the LTS console, click the name of the corresponding log stream. On the **Raw Logs** page, you can view access logs.

The following figure shows the log format, which cannot be changed.

Figure 10-81 Log format



10.6 Performing Operations in Review Center

On the **Review Center** page, API developers and callers can review the requests for operations such as publishing APIs.

APIs must be reviewed and approved before they can be published. APIs are reviewed in the following way:

- An API publisher who does not have the reviewer permission must submit the API to the reviewer for review.

- An API publisher who has the reviewer permission can publish an API without review or approval.

Requests can also be withdrawn on the **Review Center** page.

NOTE

An admin, developer, or operator can be a reviewer. A viewer cannot be a reviewer.

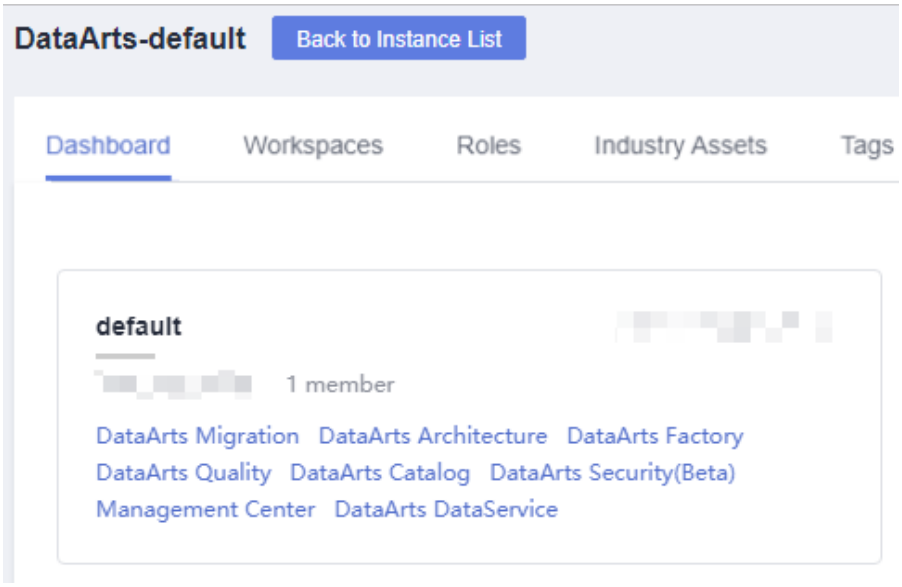
Regardless of whether they are added as reviewers, users with the admin role of the workspace have the reviewer permissions by default.

Managing a Reviewer

You can create and delete reviewers. The following procedure describes how to create a reviewer.

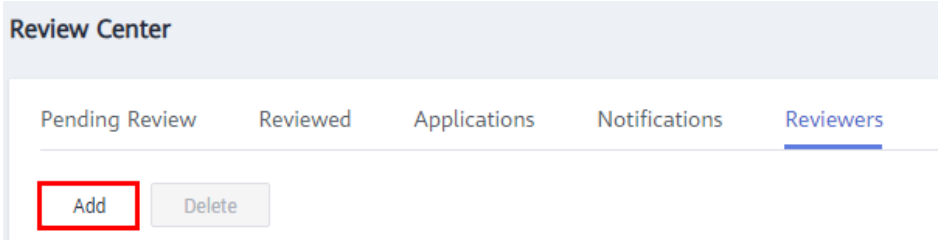
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-82 DataArts DataService



2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **Operation Management** > **Review Center** from the left navigation pane. On the page displayed, choose **Reviewer Management** and click **Add**.

Figure 10-83 Adding reviewers

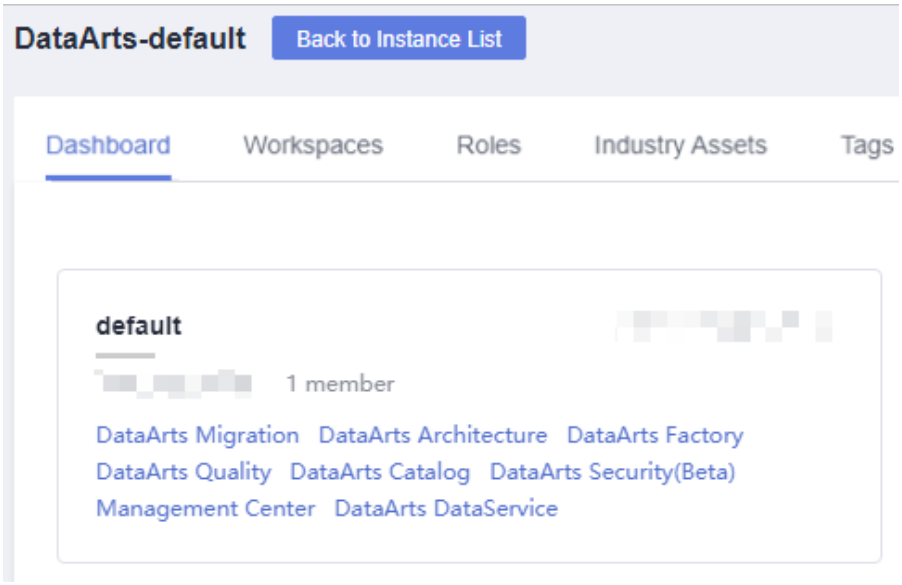


4. Select a reviewer (workspace member), enter a correct phone number and email address, and click **OK**.
5. Add more reviewers, if required.

Reviewing an Application

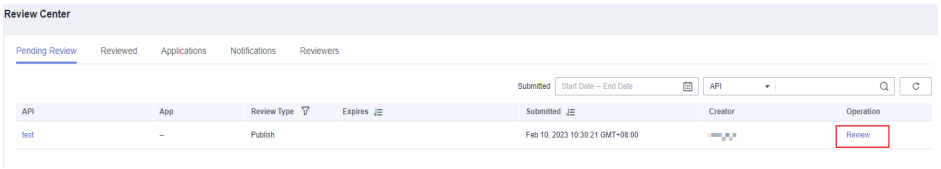
1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-84 DataArts DataService



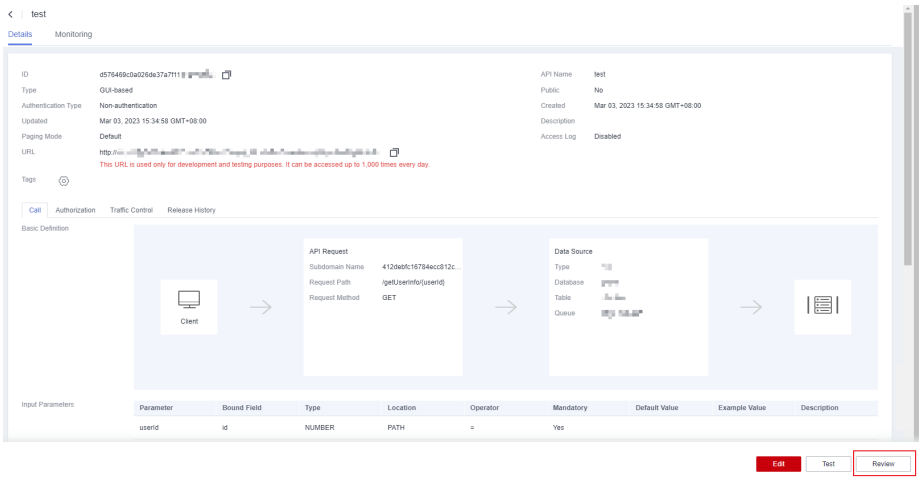
2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **Operation Management** > **Review Center** in the left navigation bar and click the **Pending Review** tab.
4. Locate the task you want to review based on the search criteria such as the review type and submission time. Then, select **Review** in the **Operation** column.

Figure 10-85 Review



You can also click the API name to go to its details page and click **Review** in the lower right corner.

Figure 10-86 Review

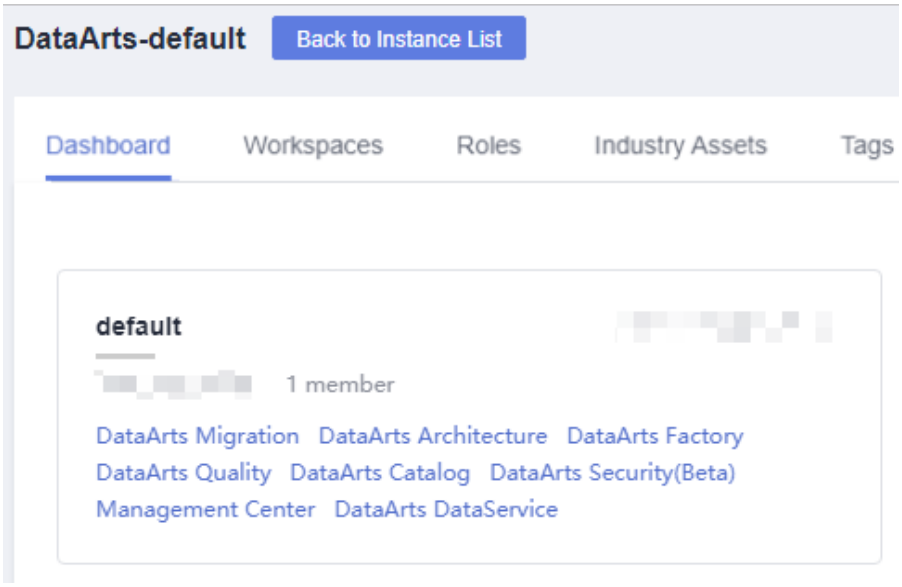


Canceling an API Application

DataArts DataService provides the function of canceling applications to be reviewed. You can cancel applications to be reviewed on the **Applications** tab page on the **Review Center** page.

1. On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts DataService**.

Figure 10-87 DataArts DataService



2. In the left navigation pane, choose an edition, for example, **Exclusive Edition**. The **Overview** page is displayed.
3. Choose **Operations Management** > **Review Center** in the left navigation pane and click the **Applications** tab.

4. Locate the row that contains the API to be canceled, and click **Cancel** in the **Operation** column.

11 Audit Log

11.1 Viewing Traces

Overview

You can use Cloud Trace Service (CTS) to record key operation events related to DataArts Studio. The events can be used in various scenarios such as security analysis, compliance audit, resource tracing, and problem locating.

After you enable CTS, the system starts to record DataArts Studio operations. The management console of CTS stores the traces of the latest seven days.

Prerequisites

CTS has been enabled. For details about how to enable it, see **Enabling CTS**.

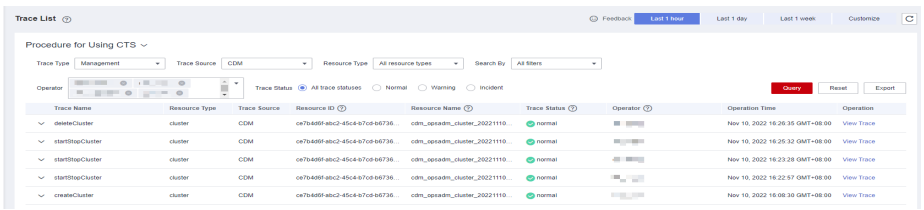
Procedure

1. Log in to the management console and choose **Cloud Trace Service** from the service list.
2. The trace list is displayed by default. You can filter traces.

   The sources of DataArts Studio traces include:

   - **CDM**: traces of DataArts Migration
   - **DLF**: traces of DataArts Factory
   - **DLG**: traces of Management Center, DataArts Architecture, DataArts Quality, DataArts Catalog, and DataArts DataService

Figure 11-1 CDM traces



3. Click ⌄ on the left of a trace to expand its details.
4. Click **View Trace** in the **Operation** column to view the trace structure details.
   For more information about CTS, see ***Cloud Trace Service User Guide***.

11.2 Key Operations Recorded by CTS

11.2.1 Management Center Operations

CTS provides records of operations on cloud service resources. With CTS, you can query, audit, and backtrack those operations.

Table 11-1 Key operations recorded by CTS

Operation	Resource Type	Trace Name
Creating data connections	dataWarehouse	createDataWarehouse
Editing data connections	dataWarehouse	updateDataWarehouse
Deleting data connections	dataWarehouse	deleteDataWarehouse
Creating workspaces	workspace	createWorkspaces
Updating workspaces	workspace	updateWorkspaces
Deleting workspaces	workspace	deleteWorkspaces
Freezing workspaces	workspace	frozenWorkspaces
Unfreezing workspaces	workspace	unfrozenWorkspaces
Adding workspace users	User	saveWorkspaceUser
Editing workspace users	User	updateWorkspaceUser
Deleting workspace users	User	deleteWorkspaceUser
Downloading files	Config	downloadFile
Creating import/export tasks	Config	createObsImportOrExport-Task

11.2.2 DataArts Migration Operations

CTS provides records of operations on cloud service resources. With CTS, you can query, audit, and backtrack those operations.

Table 11-2 Key operations recorded by CTS

Operation	Resource Type	Trace Name
Creating a cluster	cluster	createCluster
Deleting a cluster	cluster	deleteCluster
Modifying cluster configurations	cluster	modifyCluster
Starting a cluster	cluster	startCluster
Restarting a cluster	cluster	restartCluster
Importing a job	cluster	clusterImportJob
Binding an EIP	cluster	bindEip
Unbinding an EIP	cluster	unbindEip
Creating a link	link	createLink
Modifying a link	link	modifyLink
Testing a link	link	verifyLink
Deleting a link	link	deleteLink
Creating a job	job	createJob
Modifying a job	job	modifyJob
Deleting a job	job	deleteJob
Starting a job	job	startJob
Stopping a job	job	stopJob

11.2.3 DataArts Architecture Operations

CTS provides records of operations on cloud service resources. With CTS, you can query, audit, and backtrack those operations.

Table 11-3 Key operations recorded by CTS

Operation	Resource Type	Resource Names	Trace Name
Querying subjects	DAYU_DS	dsSubject	getListSubject
Creating subjects	DAYU_DS	dsSubject	createSubject
Updating subjects	DAYU_DS	dsSubject	updateSubject

Operation	Resource Type	Resource Names	Trace Name
Publishing subjects	DAYU_DS	dsSubject	publishedSubject
Suspending subjects	DAYU_DS	dsSubject	offlineSubject
Deleting subjects	DAYU_DS	dsSubject	deleteSubject
Querying processes	DAYU_DS	dsBizCatalog	getListBizCatalog
Creating processes	DAYU_DS	dsBizCatalog	createBizCatalog
Updating processes	DAYU_DS	dsBizCatalog	updateBizCatalog
Deleting processes	DAYU_DS	dsBizCatalog	deleteBizCatalog
Querying lookup tables	DAYU_DS	dsCodeTable	getListCodeTable
Creating lookup tables	DAYU_DS	dsCodeTable	createCodeTable
Updating lookup tables	DAYU_DS	dsCodeTable	updateCodeTable
Publishing lookup tables	DAYU_DS	dsCodeTable	publishedCodeTable
Suspending lookup tables	DAYU_DS	dsCodeTable	offlineCodeTable
Deleting lookup tables	DAYU_DS	dsCodeTable	deleteCodeTable
Querying data standards	DAYU_DS	dsStandardElement	getListStandardElement
Creating data standards	DAYU_DS	dsStandardElement	createStandardElement
Updating data standards	DAYU_DS	dsStandardElement	updateStandardElement
Publishing data standards	DAYU_DS	dsStandardElement	publishedStandardElement

Operation	Resource Type	Resource Names	Trace Name
Suspending data standards	DAYU_DS	dsStandardElement	offlineStandardElement
Deleting data standards	DAYU_DS	dsStandardElement	deleteStandardElement
Querying logical entities or physical tables	DAYU_DS	dsTableModel	getListTableModel
Creating logical entities or physical tables	DAYU_DS	dsTableModel	createTableModel
Updating logical entities or physical tables	DAYU_DS	dsTableModel	updateTableModel
Publishing logical entities or physical tables	DAYU_DS	dsTableModel	publishedTableModel
Suspending logical entities or physical tables	DAYU_DS	dsTableModel	offlineTableModel
Deleting logical entities or physical tables	DAYU_DS	dsTableModel	deleteTableModel
Querying dimensions	DAYU_DS	dsDimension	getListDimension
Creating dimensions	DAYU_DS	dsDimension	createDimension
Updating dimensions	DAYU_DS	dsDimension	updateDimension

Operation	Resource Type	Resource Names	Trace Name
Publishing dimensions	DAYU_DS	dsDimension	publishedDimension
Suspending dimensions	DAYU_DS	dsDimension	offlineDimension
Deleting dimensions	DAYU_DS	dsDimension	deleteDimension
Querying dimension tables	DAYU_DS	dsDimensionLogicTable	getListDimensionLogicTable
Deleting dimension tables	DAYU_DS	dsDimensionLogicTable	deleteDimensionLogicTable
Querying fact tables	DAYU_DS	dsFactLogicTable	getListFactLogicTable
Creating fact tables	DAYU_DS	dsFactLogicTable	createFactLogicTable
Updating fact tables	DAYU_DS	dsFactLogicTable	updateFactLogicTable
Publishing fact tables	DAYU_DS	dsFactLogicTable	publishedFactLogicTable
Suspending fact tables	DAYU_DS	dsFactLogicTable	offlineFactLogicTable
Deleting fact tables	DAYU_DS	dsFactLogicTable	deleteFactLogicTable
Querying summary tables	DAYU_DS	dsAggregationLogicTable	getListAggregationLogicTable
Creating summary tables	DAYU_DS	dsAggregationLogicTable	createAggregationLogicTable
Updating summary tables	DAYU_DS	dsAggregationLogicTable	updateAggregationLogicTable
Publishing summary tables	DAYU_DS	dsAggregationLogicTable	publishedAggregationLogicTable
Suspending summary tables	DAYU_DS	dsAggregationLogicTable	offlineAggregationLogicTable

Operation	Resource Type	Resource Names	Trace Name
Deleting summary tables	DAYU_DS	dsAggregationLogicTable	deleteAggregationLogicTable
Querying business metrics	DAYU_DS	dsBizMetric	getListBizMetric
Creating business metrics	DAYU_DS	dsBizMetric	createBizMetric
Updating business metrics	DAYU_DS	dsBizMetric	updateBizMetric
Publishing business metrics	DAYU_DS	dsBizMetric	publishedBizMetric
Suspending business metrics	DAYU_DS	dsBizMetric	offlineBizMetric
Deleting business metrics	DAYU_DS	dsBizMetric	deleteBizMetric
Querying atomic metrics	DAYU_DS	dsAtomicIndex	getListAtomicIndex
Creating atomic metrics	DAYU_DS	dsAtomicIndex	createAtomicIndex
Updating atomic metrics	DAYU_DS	dsAtomicIndex	updateAtomicIndex
Publishing atomic metrics	DAYU_DS	dsAtomicIndex	publishedAtomicIndex
Suspending atomic metrics	DAYU_DS	dsAtomicIndex	offlineAtomicIndex
Deleting atomic metrics	DAYU_DS	dsAtomicIndex	deleteAtomicIndex

Operation	Resource Type	Resource Names	Trace Name
Querying derivative metrics	DAYU_DS	dsDerivativeIndex	getListDerivativeIndex
Creating derivative metrics	DAYU_DS	dsDerivativeIndex	createDerivativeIndex
Updating derivative metrics	DAYU_DS	dsDerivativeIndex	updateDerivativeIndex
Deleting derivative metrics	DAYU_DS	dsDerivativeIndex	deleteDerivativeIndex
Publishing derivative metrics	DAYU_DS	dsDerivativeIndex	publishedDerivativeIndex
Suspending derivative metrics	DAYU_DS	dsDerivativeIndex	offlineDerivativeIndex
Querying compound metrics	DAYU_DS	dsCompoundMetric	getListCompoundMetric
Creating compound metrics	DAYU_DS	dsCompoundMetric	createCompoundMetric
Updating compound metrics	DAYU_DS	dsCompoundMetric	updateCompoundMetric
Deleting compound metrics	DAYU_DS	dsCompoundMetric	deleteCompoundMetric
Publishing compound metrics	DAYU_DS	dsCompoundMetric	publishedCompoundMetric
Suspending compound metrics	DAYU_DS	dsCompoundMetric	offlineCompoundMetric
Querying time filters	DAYU_DS	dsTimeCondition	getListTimeCondition
Creating time filters	DAYU_DS	dsTimeCondition	createTimeCondition

Operation	Resource Type	Resource Names	Trace Name
Updating time filters	DAYU_DS	dsTimeCondition	updateTimeCondition
Publishing time filters	DAYU_DS	dsTimeCondition	publishedTimeCondition
Suspending time filters	DAYU_DS	dsTimeCondition	offlineTimeCondition
Deleting time filters	DAYU_DS	dsTimeCondition	deleteTimeCondition
Querying directories	DAYU_DS	dsDirectory	getListDirectory
Creating directories	DAYU_DS	dsDirectory	createDirectory
Updating directories	DAYU_DS	dsDirectory	updateDirectory
Deleting directories	DAYU_DS	dsDirectory	deleteDirectory
Querying models	DAYU_DS	dsModel	getListModel
Creating models	DAYU_DS	dsModel	createModel
Updating models	DAYU_DS	dsModel	updateModel
Deleting models	DAYU_DS	dsModel	deleteModel

11.2.4 DataArts Factory Operations

CTS provides records of operations on cloud service resources. With CTS, you can query, audit, and backtrack those operations.

Table 11-4 Key operations recorded by CTS

Operation	Resource Type	Trace Name
Creating a job	job	createJob(api)
Modifying a job	job	editJob(api)
Saving a job	job	saveJob
Deleting a job	job	deleteJob

Operation	Resource Type	Trace Name
Renaming a job	job	renameJob
Importing a job	job	importPipeline/ importJob(api)
Exporting a job	job	exportPipeline/ exportJob(api)
Exporting jobs	job	exportJobs(api)
Submitting a job version	job	addNewVersion
Locking a job	job	acquireEditLock
Unlocking a job	job	releaseLock
Unlocking jobs	job	batchReleaseEditLock
Testing a job	job	testRun
Starting a job	job	startJob
Starting a job with a specified name	job	startJobByName
Stopping a job	job	stopJob
Stopping jobs	job	stopJobs
Pausing a job	job	pauseJob
Copying and saving a job	job	copyAndSaveJob
Deleting jobs	job	deleteDirectoryList
Moving a job	job	move
Stopping an instance	task	stopTask/stop(api)
Forcibly making the execution of an instance succeed	task	forceTaskSuccess
Continuing to execute an instance	task	continueExecute
Rerunning an instance	task	retryTask/restart(api)
Pausing a node	task	pauseJob
Resuming a node	task	resumeJob
Retrying a node	task	redoJobs
Skipping a node	task	skipJob

Operation	Resource Type	Trace Name
Forcibly making the execution of a node succeed	task	forceJobSuccess
Creating a script	script	addScript/createScript(api)
Executing a script	script	executeScript
Modifying a script	script	saveScript/editScript(api)
Exporting a script	script	exportScripts
Importing a script	script	importScript
Checking the syntax of a script	script	checkSyntax
Submitting a script version	script	addNewVersion
Locking a script	script	acquireScriptLock
Unlocking a script	script	releaseScriptLock
Unlocking scripts	script	batchReleaseScriptLock
Deleting scripts	script	deleteDirectoryList
Moving a script	script	move
Creating a directory	directory	createDirectory
Modifying a directory	directory	modifyDirectory
Deleting a directory	directory	deleteDirectoryByPath
Moving a directory	directory	move
Deleting directories	directory	deleteDirectoryList
Creating a data connection	dataWarehouse	createDataWarehouse
Testing a data connection	dataWarehouse	testDataWarehouseConnec-tivity
Updating a data connection	dataWarehouse	updateDataWarehouse
Deleting a data connection	dataWarehouse	deleteDataWarehouse
Exporting a data connection	dataWarehouse	exportConnection
Importing a data connection	dataWarehouse	importConnection

Operation	Resource Type	Trace Name
Creating a database	dataWarehouse	createDatabase
Updating a database	dataWarehouse	updateDatabase
Deleting a database	dataWarehouse	deleteDatabase
Creating a data table	dataWarehouse	createDataTable
Updating a data table	dataWarehouse	updateDataTable
Deleting a data table	dataWarehouse	deleteDataTable
Creating a schema	dataWarehouse	createSchema
Deleting a schema	dataWarehouse	deleteSchema
Updating a schema	dataWarehouse	updateSchema
Create a notification	alarmRule	createAlarmRules
Creating and updating a notification	alarmRule	createAndUpdateAlarm-Rules
Deleting a notification	alarmRule	deleteAlarmRules
Updating a notification	alarmRule	updateAlarmRules
Creating a resource	dataResource	createResource
Updating a resource	dataResource	updateResource
Deleting a resource	dataResource	deleteResources
Exporting a resource	dataResource	exportResource
Importing a resource	dataResource	importResource
Deleting resources	dataResource	deleteDirectoryList
Creating a tag	tag	create
Deleting a tag	tag	delete
Exporting a tag	tag	exportJobTags
Import a tag from OBS	tag	importJobTag
Importing a tag from a local server	tag	importJobTag2
Saving an environment variable	environmentVariable	saveEnvParams

Operation	Resource Type	Trace Name
Deleting an environment variable	environmentVariable	deleteEnvParams
Exporting an environment variable	environmentVariable	exportEnvParams
Importing an environment variable	environmentVariable	importEnvParams
Updating a workspace configuration item	workspaceConfig	updateWorkSpaceConfigs
Uploading a file	file	uploadFile
Configuring an agency	agency	saveAgency
Saving a sensitive variable	sensitiveParam	saveSensitiveParam
Updating a sensitive variable	sensitiveParam	updateSensitiveParam
Deleting a sensitive variable	sensitiveParam	deleteSensitiveParam
Creating a CDM connection	createConnection	cdmConnection
Updating a CDM connection	updateConnection	cdmConnection
Deleting a CDM connection	deleteConnection	cdmConnection
Sending an HTTP trigger message	sendMessage	httpTriggerMessage

11.2.5 DataArts Quality Operations

CTS provides records of operations on cloud service resources. With CTS, you can query, audit, and backtrack those operations.

Table 11-5 Key operations recorded by CTS

Operation	Resource Type	Trace Name
Creating directories	Category	createCategory
Deleting directories	Category	deleteCategory
Updating directories	Category	updateCategory

Operation	Resource Type	Trace Name
Batch stopping instances	Instance	batchStop
Batch deleting instances	Instance	batchDeleteInstances
Creating comparison jobs	ConsistencyTask	createConsistencyTask
Deleting comparison jobs	ConsistencyTask	batchDeleteConsistencyTask
Editing comparison jobs	ConsistencyTask	editConsistencyTask
Starting scheduling comparison jobs	ConsistencyTask	startScheduleConsistency-Task
Stopping scheduling comparison jobs	ConsistencyTask	stopScheduleConsistency-Task
Running comparison jobs	ConsistencyTask	runConsistencyTask
Creating quality jobs	Rule	createRuleTask
Deleting quality jobs	Rule	deleteRule
Updating quality jobs	Rule	updateRule
Running a quality job	Rule	instanceScheduleOperation
Running quality jobs	Rule	batchInstanceScheduleOp-eration
Operating quality jobs	Rule	batchOperateRules
Creating rule templates	RuleTemplate	createTemplate
Deleting rule templates	RuleTemplate	deleteTemplate
Querying rule templates	RuleTemplate	getRuleTemplateList
Updating rule templates	RuleTemplate	updateTemplate
Querying a rule template	RuleTemplate	getTemplate
Obtaining the quality jobs and comparison jobs that depend on rule templates	RuleTemplate	getDependentTasks

Operation	Resource Type	Trace Name
Updating rule templates for jobs	RuleTemplate	batchUpdateDependent-Tasks

11.2.6 DataArts Catalog Operations

CTS provides records of operations on cloud service resources. With CTS, you can query, audit, and backtrack those operations.

Table 11-6 Key operations recorded by CTS

Operation	Resource Type	Trace Name
Creating data masks	datamask	createDataMask
Querying data masks	datamask	listDataMask
Querying a data mask	datamask	getDataMask
Deleting a data mask	datamask	deleteDataMask
Deleting data masks	datamask	batchDeleteDataMask
Updating data masks	datamask	updateDataMask
Creating and running a collection task	bridgetask	createBridgeTask
Querying collection tasks	bridgetask	getBridgeTask
Editing collection tasks	bridgetask	updateBridgeTask
Deleting collection tasks	bridgetask	batchDeleteBridgeTask
Adding a tag to a data asset	asset	addTagToAsset
Adding a tag	tag	createTag
Adding tags	tag	batchCreateTag
Deleting tags	tag	batchDeleteTag
Updating a tag	tag	updateTag
Querying tags	tag	getTags
Deleting a tag	tag	deleteTag
Creating a task directory	bridgetaskcategory	createBridgeTaskCategory

Operation	Resource Type	Trace Name
Obtaining task directories	bridgetaskcategory	getBridgeTaskCategoryTree
Editing a task directory	bridgetaskcategory	updateBridgeTaskCategory
Deleting a task directory	bridgetaskcategory	deleteBridgeTaskCategory
Creating a classification group	classificationgroup	createClassificationGroup
Querying classification groups	classificationgroup	listClassificationGroup
Querying a classification group	classificationgroup	getClassificationGroup
Deleting classification groups	classificationgroup	batchDeleteClassifica-tionGroup
Modifying a classification group	classificationgroup	updateClassificationGroup
Creating a classification rule	classificationrule	createClassificationRule
Querying classification rules	classificationrule	listClassificationRule
Querying a classification rule	classificationrule	getClassificationRule
Deleting classification rules	classificationrule	batchDeleteClassification-Rule
Modifying a classification rule	classificationrule	updateClassificationRule
Creating a data security level	secrecylevel	createSecrecyLevel
Querying data security levels	secrecylevel	listSecrecyLevel
Querying a data security level	secrecylevel	getSecrecyLevel
Deleting data security levels	secrecylevel	batchDeleteSecrecyLevel
Modifying a data security level	secrecylevel	updateSecrecyLevel
Creating collection tasks	bridgetask	createBridgeTask

Operation	Resource Type	Trace Name
Editing collection tasks	bridgetask	updateBridgeTask
Deleting collection tasks	bridgetask	deleteBridgeTask
Querying collection tasks	bridgetask	getTasks

11.2.7 DataArts DataService Operations

CTS provides records of operations on cloud service resources. With CTS, you can query, audit, and backtrack those operations.

Table 11-7 Key operations recorded by CTS

Operation	Resource Type	Trace Name
Creating an API	DLMApi	createApi
Updating an API	DLMApi	updateApi
Querying an API	DLMApi	getApi
Querying APIs	DLMApi	getApiList(Api)
Deleting an API	DLMApi	deleteApi
Publishing an API	DLMApi	publishApi
Unpublishing an API	DLMApi	unpublishApi
Renewing an API	DLMApi	renewApi
Suspending an API	DLMApi	stopApi
Restoring an API	DLMApi	recoverApi
Copying an API	DLMApi	copyApi
Operating an API	DLMApi	actionApi
Creating an app	DLMApp	createApp
Updating an app	DLMApp	updateApp
Deleting an app	DLMApp	deleteApp
Querying an app	DLMApp	getApp
Querying app details	DLMApp	getAppInfo
Authorizing an app to access an API	DLMRelation	authorizeApi

Operation	Resource Type	Trace Name
Querying authorized apps	DLMRelation	getAuthorizeApp
Canceling authorization	DLMRelation	cancelApprovalApi
Querying unauthorized apps	DLMRelation	getLeftApp
Applying for an API	DLMApply	applyApi
Canceling an application	DLMApply	revokeApply
Obtaining applications	DLMApply	getApplyList
Obtaining application details	DLMApply	getApplyDetail
Obtaining notification details	DLMApply	getMessageDetail
Creating an application	DLMApply	createApply
Reviewing applications	DLMApply	batchApproveNewApply
Sending a notification	DLMApply	sendMesg
Obtaining notifications	DLMApply	getMessageList
Obtaining the publication trend	DLMApply	getPublishTrend
Creating a throttling policy	DLMFlowControl	createFlowControlStrategy
Updating a throttling policy	DLMFlowControl	updateFlowControlStrategy
Deleting a throttling policy	DLMFlowControl	deleteFlowControlStrategy
Querying a throttling policy	DLMFlowControl	getFlowControlStrategy
Querying APIs (related to throttling)	DLMFlowControlBindApi	getAllApiList
Querying the APIs that have been associated with a throttling policy	DLMFlowControlBindApi	getBindingApiList
Associating a throttling policy with an API	DLMFlowControlBindApi	bindingApi

Operation	Resource Type	Trace Name
Disassociating a throttling policy from an API	DLMFlowControlBindApi	unBindingApi
Querying the API overview	DLMRequestRecord	getApisOverview
Querying the app overview	DLMRequestRecord	getAppsOverView
Querying top N services called by an API	DLMRequestRecord	getApisTop
Querying the top N services used by an app	DLMRequestRecord	getAppsTop
Querying API statistics details	DLMRequestRecord	getApisDetail
Querying app statistics details	DLMRequestRecord	getAppsDetail
Querying API dashboard data details	DLMRequestRecord	getApisDashboard
Querying app dashboard data details	DLMRequestRecord	getAppsDashboard
Querying top N abnormal API calls	DLMRequestRecord	getApisError
Querying supported data source types	DLMDataSourceType	getDatasources
Querying data connections	DLMDataSourceConnection	getDatasourceConnections
Querying databases	DLMDataSourceDatabase	getDatasourcedatabases
Querying data tables	DLMDataSourceTable	getDatasourcedatables
Querying table fields	DLMDataSourceTableField	getDatasourceTableFields
Querying queues	DLMDataSourceQueue	getQueue
Querying users who can be reviewers	DLMAuthorizedUser	getAuthorizedUser
Creating reviewers	DLMApprover	createApprover
Deleting reviewers	DLMApprover	deleteApprover
Querying reviewers	DLMApprover	getApproverList

Operation	Resource Type	Trace Name
Querying the content in a service catalog	DLMServiceCatalog	getCatalogAllDetail
Querying APIs in a service catalog	DLMServiceCatalog	getCatalogApis
Querying sub-catalogs in a service catalog	DLMServiceCatalog	getCatalogCatalogs
Creating service catalogs	DLMServiceCatalog	createCatalog
Deleting service catalogs	DLMServiceCatalog	deleteCatalog
Updating service catalogs	DLMServiceCatalog	updateCatalog
Querying service catalog details	DLMServiceCatalog	getCatalogDetail
Moving service catalogs	DLMServiceCatalog	moveCatalog
Moving APIs	DLMServiceCatalog	moveApi
Obtaining tags	DLMTag	getTags
Obtaining local tags	DLMTag	getLocalTags
Updating tags	DLMTag	updateTags

