DataArts Studio

Getting Started

Issue	01
Date	2023-12-11

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions

Notice

The purchased products, services and features are stipulated by the contract made between Huawei Cloud 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.

Huawei Cloud Computing Technologies Co., Ltd.

Address: Huawei Cloud Data Center Jiaoxinggong Road
Qianzhong Avenue
Gui'an New District
Gui Zhou 550029
People's Republic of China

Website: https://www.huaweicloud.com/intl/en-us/

Contents

1 Overview

DataArts Studio is a one-stop data governance platform that provides full data lifecycle management and intelligent data management capabilities. It is built on a data lake foundation and provides data integration, development, and governance capabilities. The following table lists the use cases for different application scenarios.

Table 1-1 Use cases

Example Use Case	Data Lake Foundation	Capability	Scenario
Starters: Data Development Based on E-commerce BI Reports	DLI	Data development	Full data lifecycle management is usually not required, and fully managed big data scheduling capabilities are required. Such scenarios include trial use for developers and small-scale verification.
Beginners: Data Integration and Development Based on Movie Scores	DWS	Data integration and development	Big data development scenarios such as management of data ETL tasks where data governance is not required. Such scenarios include trial use for developers and small-scale verification.
(Coming Soon) Intermediate Users: Lightweight Data Governance Based on the Analysis of Sales per Unit Area	DLI	Data integration, data development, and lightweight data governance	Lightweight data governance capabilities are required, such as DataArts Quality, DataArts Catalog, and DataArts DataService, but DataArts Architecture is not required.

Example Use Case	Data Lake Foundation	Capability	Scenario
Advanced Users: Data Governance Based on Taxi Trip Data	MRS Hive	Data integration, data development, and data governance	All data governance capabilities are required. Users should have a data management team and system and want to implement enterprise information architecture, data standards, data models, and data metrics. The DAYU data governance methodology applies to this scenario.

2 Starters: Data Development Based on E-commerce BI Reports

2.1 Scenario

In this practice, the DataArts Factory module of DataArts Studio and Data Lake Insight (DLI) are used to analyze the users, products, and anonymized comments of an e-commerce store to generate the data features of users and commodities, providing valuable information for marketing decision-making, advertising, credit rating, brand monitoring, and user behavior prediction. You will learn DataArts Factory functions such as script editing, job editing, and job scheduling, as well as basic SQL syntax of DLI.

NOTE

This practice involves the Management Center and DataArts Factory modules of DataArts Studio. All DataArts Studio versions can meet requirements.

The procedure is as follows:

1. Make preparations, including **Preparations Before Using DataArts Studio**, **preparing data sources**, and **preparing a data lake**.
2. Develop data, including creating DLI SQL scripts and a job.
   - **Analyze 10 products users like most**.
   - **Analyze 10 products users dislike most**.
   - **Develop and schedule a job**. After orchestrating the job and configuring job scheduling policies to periodically execute the job, you can obtain the latest data analysis result every day.
3. **Unsubscribe from services**. If you do not want to use DataArts Studio and related services, unsubscribe from them and delete resources in a timely manner.

2.2 Step 1: Make Preparations

Preparations Before Using DataArts Studio

If you use DataArts Studio for the first time, register a , buy a DataArts Studio instance, and create a workspace by following the instructions provided in **Preparations**. Then you can go to the created workspace and start using DataArts Studio.

Preparing Data Sources

This practice analyzes the data features of the users and products of an e-commerce store. (The data is from BI reports.)

To facilitate demonstration, this practice provides some data used to simulate the original data. To integrate the source data into the cloud, you need to store the sample data in CSV files and upload them to an OBS bucket.

Step 1 Create CSV files (UTF-8 without BOM), name the files with the corresponding data table names, copy the sample data to different CSV files, and save the files.

To generate a CSV file in Windows, you can perform the following steps:

1. Use a text editor (for example, Notepad) to create a .txt document and copy the sample data to the document. Then check the total number of rows and check whether the data of rows is correctly separated. (If the sample data is copied from a PDF document, the data in a single row will be wrapped if the data is too long. In this case, you must manually adjust the data to ensure that it is in a single row.)
2. Choose **File** > **Save as**. In the displayed dialog box, set **Save as type** to **All files (*.*)**, enter the file name with the .csv suffix for **File name**, and select the UTF-8 encoding format (without BOM) to save the file in CSV format.

Step 2 Upload the CSV file to OBS.

1. Log in to the management console and choose **Storage** > **Object Storage Service** to access the OBS console.
2. Click **Create Bucket** and set parameters as prompted to create an OBS bucket named **fast-demo**.

   NOTE

   To ensure network connectivity, select the same region for OBS bucket as that for the DataArts Studio instance. If an enterprise project is required, select the enterprise project that is the same as that of the DataArts Studio instance.

   For details about how to create a bucket on the OBS console, see **Creating a Bucket** in *Object Storage Service Console Operation Guide*.
3. Upload data to OBS bucket **fast-demo**.

   For details about how to upload a file on the OBS console, see **Uploading a File** in *Object Storage Service Console Operation Guide*.

----End

This practice involves the following sample data: user data (**user_data.csv**), product data (**product_data.csv**), comment data (**comment_data.csv**), and action data (**action_data.csv**). Descriptions of the data are as follows:

- **user_data.csv**:

  ```
  user_id,age,gender,rank,register_time 100001,20,0,
  1,2021/1/1
  100002,22,1,2,2021/1/2
  100003,21,0,3,2021/1/3
  100004,24,2,5,2021/1/4
  100005,50,2,9,2021/1/5
  100006,20,1,3,2021/1/6
  100007,18,1,1,2021/1/7
  100008,20,1,6,2021/1/8
  100009,60,0,4,2021/1/9
  100010,20,1,1,2021/1/10
  100011,35,0,5,2021/1/11
  100012,20,1,1,2021/1/12
  100013,7,0,1,2021/1/13
  100014,64,0,8,2021/1/14
  100015,20,1,1,2021/1/15
  100016,33,1,7,2021/1/16
  100017,20,0,1,2021/1/17
  100018,15,1,1,2021/1/18
  100019,20,1,9,2021/1/19
  100020,33,0,1,2021/1/20
  100021,20,0,1,2021/1/21
  100022,22,1,5,2021/1/22
  100023,20,1,1,2021/1/23
  100024,20,0,1,2021/1/24
  100025,34,0,7,2021/1/25
  100026,34,1,1,2021/1/26
  100027,20,1,8,2021/1/27
  100028,20,0,1,2021/1/28
  100029,56,0,5,2021/1/29
  100030,20,1,1,2021/1/30
  100031,22,1,8,2021/1/31
  100032,20,0,1,2021/2/1
  100033,32,1,0,2021/2/2
  100034,20,1,1,2021/2/3
  100035,45,0,6,2021/2/4
  100036,20,0,1,2021/2/5
  100037,67,1,4,2021/2/6
  100038,78,0,6,2021/2/7
  100039,11,1,8,2021/2/8
  100040,8,0,0,2021/2/9
  ```

  The following table describes the data.

  Table 2-1 User data description

Field	Type	Description	Value
user_id	int	User ID	Anonymized
age	int	Age group	**-1** indicates that the user age is unknown.
gender	int	Gender	● **0**: male ● **1**: female ● **2**: confidential
rank	Int	User level	The greater the value of this field, the higher the user level.

Field	Type	Description	Value
register_time	string	User registration date	Unit: day

- **product_data.csv**:

```
product_id,a1,a2,a3,category,brand
200001,1,1,1,300001,400001
200002,2,2,2,300002,400001
200003,3,3,3,300003,400001
200004,1,2,3,300004,400001
200005,3,2,1,300005,400002
200006,1,1,1,300006,400002
200007,2,2,2,300007,400002
200008,3,3,3,300008,400002
200009,1,2,3,300009,400003
200010,3,2,1,300010,400003
200011,1,1,1,300001,400003
200012,2,2,2,300002,400003
200013,3,3,3,300003,400004
200014,1,2,3,300004,400004
200015,3,2,1,300005,400004
200016,1,1,1,300006,400004
200017,2,2,2,300007,400005
200018,3,3,3,300008,400005
200019,1,2,3,300009,400005
200020,3,2,1,300010,400005
200021,1,1,1,300001,400006
200022,2,2,2,300002,400006
200023,3,3,3,300003,400006
200024,1,2,3,300004,400006
200025,3,2,1,300005,400007
200026,1,1,1,300006,400007
200027,2,2,2,300007,400007
200028,3,3,3,300008,400007
200029,1,2,3,300009,400008
200030,3,2,1,300010,400008
200031,1,1,1,300001,400008
200032,2,2,2,300002,400008
200033,3,3,3,300003,400009
200034,1,2,3,300004,400009
200035,3,2,1,300005,400009
200036,1,1,1,300006,400009
200037,2,2,2,300007,400010
200038,3,3,3,300008,400010
200039,1,2,3,300009,400010
200040,3,2,1,300010,400010
```

The following table describes the data.

Table 2-2 Product data description

Field	Type	Description	Value
product_id	int	Product No.	Anonymized
a1	int	Attribute 1	Enumerated value. The value **-1** indicates unknown.

Field	Type	Description	Value
a2	int	Attribute 2	Enumerated value. The value **-1** indicates unknown.
a3	int	Attribute 3	Enumerated value. The value **-1** indicates unknown.
category	int	Category ID	Anonymized
brand	int	Brand ID	Anonymized

- **comment_data.csv**:

```
deadline,product_id,comment_num,has_bad_comment,bad_comment_rate
2021/3/1,200001,4,0,0
2021/3/1,200002,1,0,0
2021/3/1,200003,2,2,0.1
2021/3/1,200004,3,3,0.05
2021/3/1,200005,1,0,0
2021/3/1,200006,2,0,0
2021/3/1,200007,3,2,0.01
2021/3/1,200008,4,1,0.001
2021/3/1,200009,4,0,0
2021/3/1,200010,1,0,0
2021/3/1,200011,2,2,0.2
2021/3/1,200012,3,3,0.04
2021/3/1,200013,1,0,0
2021/3/1,200014,2,2,0.2
2021/3/1,200015,3,2,0.05
2021/3/1,200016,4,1,0.003
2021/3/1,200017,4,0,0
2021/3/1,200018,1,0,0
2021/3/1,200019,2,2,0.3
2021/3/1,200020,3,3,0.03
2021/3/1,200021,1,0,0
2021/3/1,200022,2,5,1
2021/3/1,200023,3,2,0.07
2021/3/1,200024,4,1,0.006
2021/3/1,200025,4,0,0
2021/3/1,200026,1,0,0
2021/3/1,200027,2,2,0.4
2021/3/1,200028,3,3,0.03
2021/3/1,200029,1,0,0
2021/3/1,200030,2,5,1
2021/3/1,200031,3,2,0.02
2021/3/1,200032,4,1,0.003
2021/3/1,200033,4,0,0
2021/3/1,200034,1,0,0
2021/3/1,200035,2,2,0.5
2021/3/1,200036,3,3,0.06
2021/3/1,200037,1,0,0
2021/3/1,200038,2,1,0.01
2021/3/1,200039,3,2,0.01
2021/3/1,200040,4,1,0.009
```

The following table describes the data.

Table 2-3 Comment data description

Field	Type	Description	Value
deadline	string	Deadline	Unit: day
product_id	int	Product No.	Anonymized
comment_num	int	Segments of the accumulated comment count	● **0**: no comment ● **1**: one comment ● **2**: 2 to 10 comments ● **3**: 11 to 50 comments ● **4**: more than 50 comments
has_bad_comment	int	Whether there are negative comments	**0**: no; **1**: yes
bad_comment_rate	float	Dissatisfaction rate	Proportion of negative comments

- **action_data.csv**:

```
user_id,product_id,time,model_id,type
100001,200001,2021/1/1,1,view
100001,200001,2021/1/1,1,add
100001,200001,2021/1/1,1,delete
100001,200002,2021/1/2,1,view
100001,200002,2021/1/2,1,add
100001,200002,2021/1/2,1,buy
100001,200002,2021/1/2,1,like
100002,200003,2021/1/1,1,view
100002,200003,2021/1/1,1,add
100002,200003,2021/1/1,1,delete
100002,200004,2021/1/2,1,view
100002,200004,2021/1/2,1,add
100002,200004,2021/1/2,1,buy
100002,200004,2021/1/2,1,like
100003,200001,2021/1/1,1,view
100003,200001,2021/1/1,1,add
100003,200001,2021/1/1,1,delete
100004,200002,2021/1/2,1,view
100005,200002,2021/1/2,1,add
100006,200002,2021/1/2,1,buy
100007,200002,2021/1/2,1,like
100001,200003,2021/1/1,1,view
100002,200003,2021/1/1,1,add
100003,200003,2021/1/1,1,delete
100004,200004,2021/1/2,1,view
100005,200004,2021/1/2,1,add
100006,200004,2021/1/2,1,buy
100007,200004,2021/1/2,1,like
100001,200005,2021/1/3,1,view
100001,200005,2021/1/3,1,add
100001,200005,2021/1/3,1,delete
100001,200006,2021/1/3,1,view
100001,200006,2021/1/4,1,add
100001,200006,2021/1/4,1,buy
100001,200006,2021/1/4,1,like
100010,200005,2021/1/3,1,view
100010,200005,2021/1/3,1,add
```

```
100010,200005,2021/1/3,1,delete
100010,200006,2021/1/3,1,view
100010,200006,2021/1/4,1,add
100010,200006,2021/1/4,1,buy
100010,200006,2021/1/4,1,like
100001,200007,2021/1/2,1,buy
100001,200007,2021/1/2,1,like
100002,200007,2021/1/1,1,view
100002,200007,2021/1/1,1,add
100002,200007,2021/1/1,1,delete
100002,200007,2021/1/2,1,view
100002,200007,2021/1/2,1,add
100002,200008,2021/1/2,1,like
100002,200008,2021/1/2,1,like
100003,200008,2021/1/1,1,view
100003,200008,2021/1/1,1,add
100003,200008,2021/1/1,1,delete
100004,200008,2021/1/2,1,view
100005,200009,2021/1/2,1,like
100006,200009,2021/1/2,1,buy
100007,200010,2021/1/2,1,like
100001,200010,2021/1/1,1,view
100002,200010,2021/1/1,1,add
100003,200010,2021/1/1,1,delete
100004,200010,2021/1/2,1,view
100005,200010,2021/1/2,1,like
100006,200010,2021/1/2,1,buy
100007,200010,2021/1/2,1,like
100001,200010,2021/1/3,1,view
100001,200010,2021/1/3,1,add
100001,200010,2021/1/3,1,delete
100001,200011,2021/1/3,1,view
100001,200011,2021/1/4,1,like
100001,200011,2021/1/4,1,buy
100001,200011,2021/1/4,1,like
100010,200012,2021/1/3,1,view
100011,200012,2021/1/3,1,like
100011,200012,2021/1/3,1,delete
100011,200013,2021/1/3,1,view
100011,200013,2021/1/4,1,like
100011,200014,2021/1/4,1,buy
100011,200014,2021/1/4,1,like
100007,200022,2021/1/2,1,like
100001,200022,2021/1/1,1,view
100002,200023,2021/1/1,1,add
100003,200023,2021/1/1,1,delete
100004,200023,2021/1/2,1,like
100005,200024,2021/1/2,1,add
100006,200024,2021/1/2,1,buy
100007,200025,2021/1/2,1,like
100001,200025,2021/1/3,1,view
100001,200026,2021/1/3,1,like
100001,200026,2021/1/3,1,delete
100001,200027,2021/1/3,1,view
100001,200027,2021/1/4,1,like
100001,200027,2021/1/4,1,buy
100001,200028,2021/1/4,1,like
100010,200029,2021/1/3,1,view
100011,200030,2021/1/3,1,like
100011,200031,2021/1/3,1,delete
100011,200032,2021/1/3,1,view
100011,200033,2021/1/4,1,like
100011,200034,2021/1/4,1,buy
100011,200035,2021/1/4,1,like
```

The following table describes the data.

Table 2-4 Action data description

Field	Type	Description	Value
user_id	int	User ID	Anonymized
product_id	int	Product No.	Anonymized
time	string	Time of action	-
model_id	string	Module ID	Anonymized
type	string	● View (browsing the product details page) ● Add (adding a product to the shopping cart) ● Delete (removing a product from the shopping cart) ● Buy (placing an order) ● Like (adding a product to the favorite list)	-

Preparing a Data Lake

This practice uses DLI as the data lake. To ensure network connectivity between DataArts Studio and DLI, ensure that you select the same region and enterprise project as those of the DataArts Studio instance when creating a DLI queue.

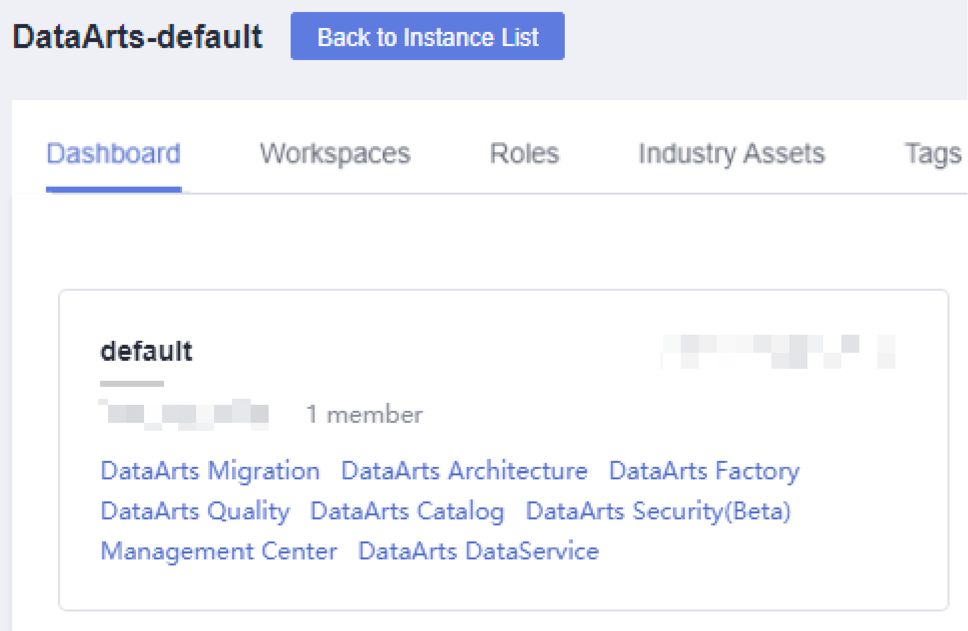
NOTE

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.

After enabling DLI, you need to create a DLI connection in Management Center, create a database through the DataArts Factory module, and run an SQL statement to create an OBS foreign table. The procedure is as follows:

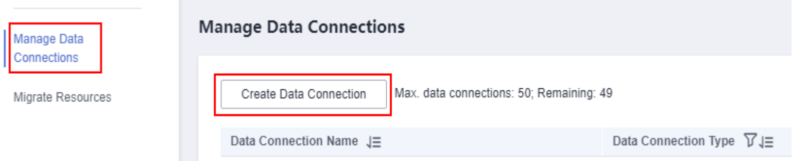
Step 1 On the DataArts Studio console, locate a workspace and click **Management Center**.

Figure 2-1 Management Center



Step 2 On the displayed **Manage Data Connections** page, click **Create Data Connection**.

Figure 2-2 Creating a data connection

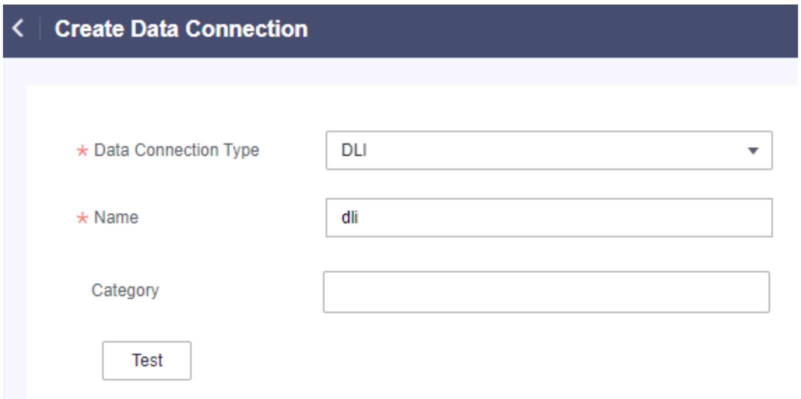


Max. data connections: 50; Remaining: 49

Step 3 Create a DLI data connection. Select **DLI** for **Data Connection Type** and set **Name** to **dli**.

Click **Test** to test the connection. If the test is successful, click **OK**.

Figure 2-3 Creating a data connection


Create Data Connection

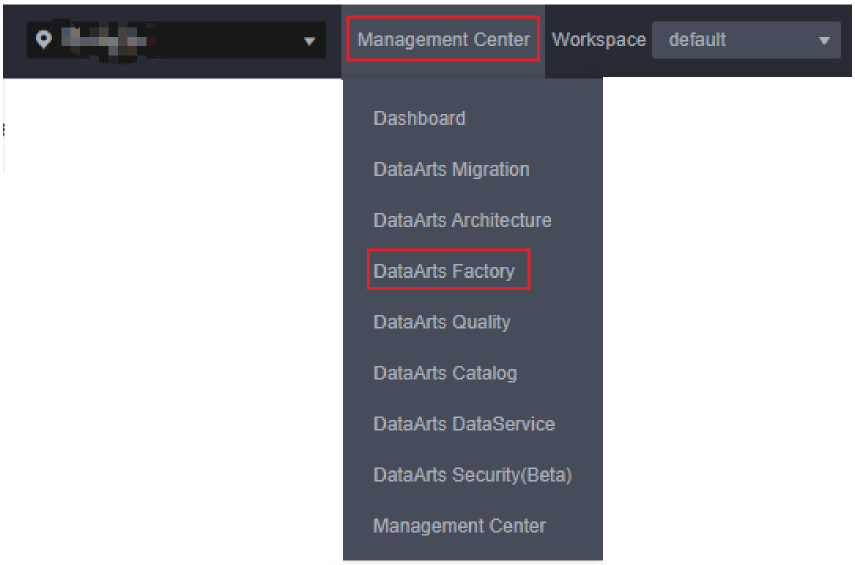
* Data Connection Type: DLI

* Name: dli

Category:

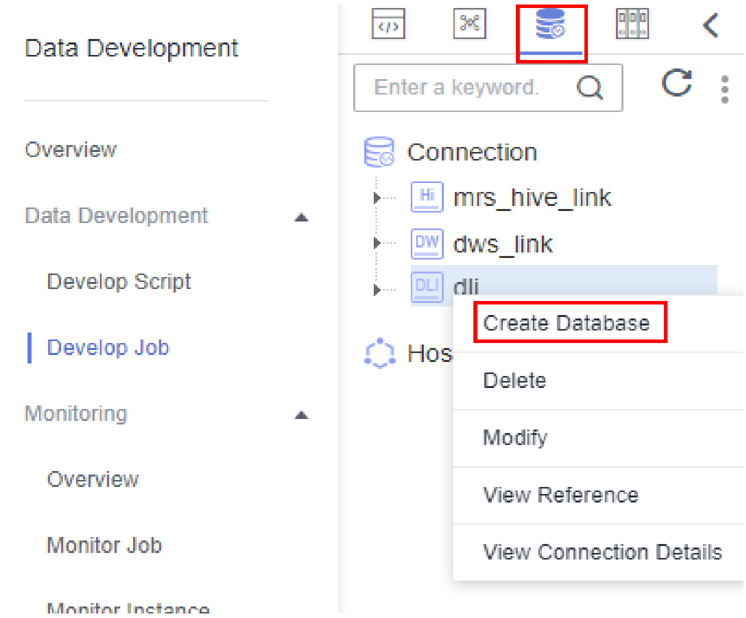
Step 4 Go to the **DataArts Factory** page.

Figure 2-4 DataArts Factory page



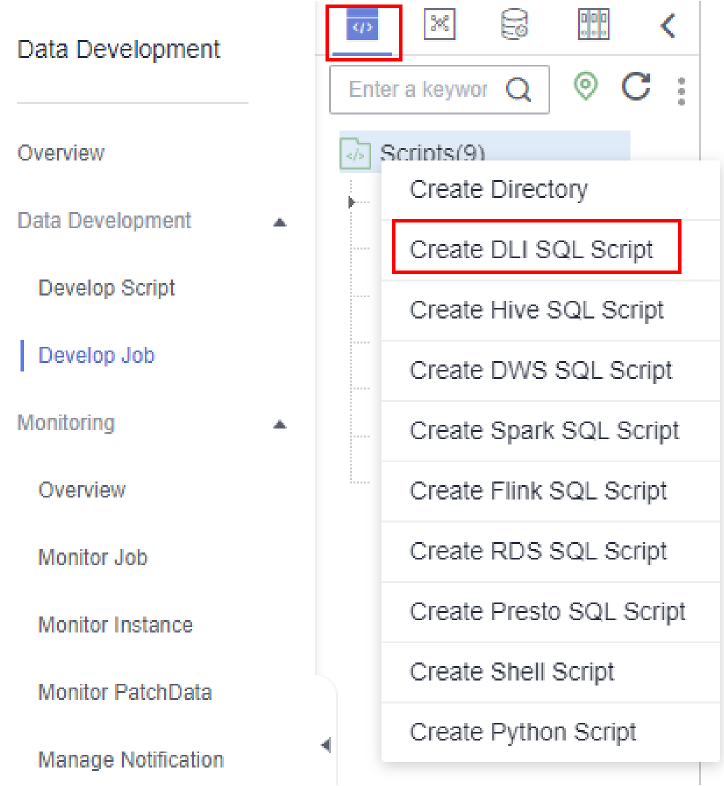
Step 5 Right-click the DLI connection to create a database named **BI** for storing data tables. For how to create a database, see **Figure 2-5**.

Figure 2-5 Creating a database



Step 6 Create a DLI SQL script used to create data tables by entering DLI SQL statements in the editor.

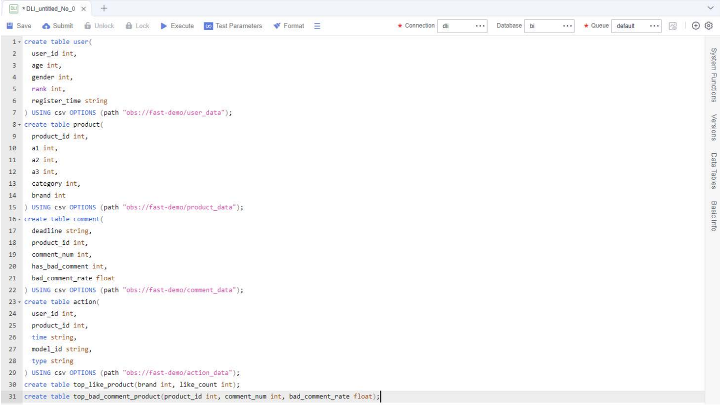
Figure 2-6 Creating a script



Step 7 In the SQL editor, enter the following SQL statements and click **Execute** to create data tables. Among them, **user**, **product**, **comment**, and **action** are OBS tables. They contain original data and are stored in OBS buckets. **top_like_product** and **top_bad_comment_product** are DLI tables that store analysis results.

```
create table user(
  user_id int,
  age int,
  gender int,
  rank int,
  register_time string
) USING csv OPTIONS (path "obs://fast-demo/user_data.csv");
create table product(
  product_id int,
  a1 int,
  a2 int,
  a3 int,
  category int,
  brand int
) USING csv OPTIONS (path "obs://fast-demo/product_data.csv");
create table comment(
  deadline string,
  product_id int,
  comment_num int,
  has_bad_comment int,
  bad_comment_rate float
) USING csv OPTIONS (path "obs://fast-demo/comment_data.csv");
create table action(
  user_id int,
  product_id int,
  time string,
  model_id string,
  type string
) USING csv OPTIONS (path "obs://fast-demo/action_data.csv");
create table top_like_product(brand int, like_count int);
create table top_bad_comment_product(product_id int, comment_num int, bad_comment_rate float);
```

Figure 2-7 Creating data tables



Connection: dli
Database: bi
Queue: default

```
create table user(
  user_id int,
  age int,
  gender int,
  rank int,
  register_time string
) USING csv OPTIONS (path "obs://fast-demo/user_data");
create table product(
  product_id int,
  a1 int,
  a2 int,
  a3 int,
  category int,
  brand int
) USING csv OPTIONS (path "obs://fast-demo/product_data");
create table comment(
  deadline string,
  product_id int,
  comment_num int,
  has_bad_comment int,
  bad_comment_rate float
) USING csv OPTIONS (path "obs://fast-demo/comment_data");
create table action(
  user_id int,
  product_id int,
  time string,
  model_id string,
  type string
) USING csv OPTIONS (path "obs://fast-demo/action_data");
create table top_like_product(brand int, like_count int);
create table top_bad_comment_product(product_id int, comment_num int, bad_comment_rate float);
```

The key parameters are as follows:

- **Data Connection**: DLI data connection created in **Step 3**
- **Database**: database created in **step 5**
- **Resource Queue**: the default resource queue **default**

Step 8 After the script is executed successfully, run the following script to check whether the data tables are created successfully.

```
SHOW TABLES;
```

NOTE

After confirming that the data tables are created, you can close the script as it is no longer needed.

----End

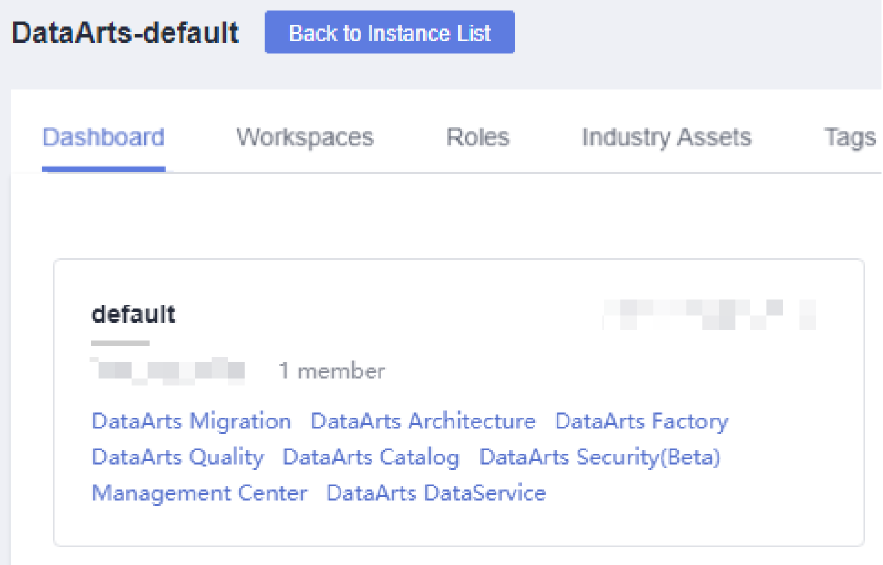
2.3 Step 2: Develop Data

This step describes how to use the data in BI reports to analyze the 10 products users like most and 10 products users dislike most. Jobs are periodically executed and the results are exported to tables every day for data analysis.

Analyze 10 Products Users Like Most

Step 1 Log in to the DataArts Studio console. Locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Factory**.

Figure 2-8 DataArts Factory


DataArts-default

Back to Instance List

Dashboard Workspaces Roles Industry Assets Tags

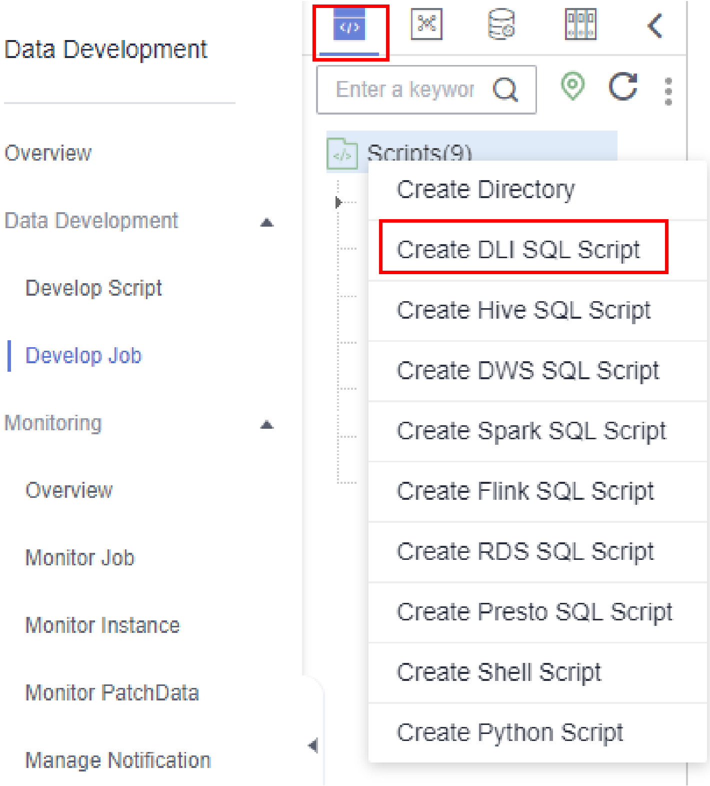
default

1 member

DataArts Migration DataArts Architecture DataArts Factory
DataArts Quality DataArts Catalog DataArts Security(Beta)
Management Center DataArts DataService

Step 2 Create a DLI SQL script used to create data tables by entering DLI SQL statements in the editor.

Figure 2-9 Creating a script

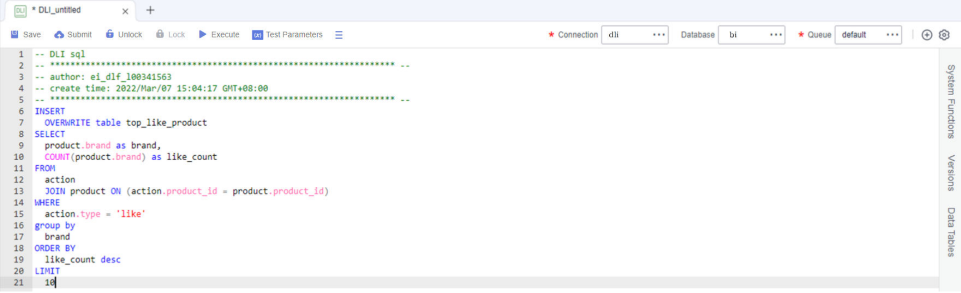


Step 3 In the SQL editor, enter the following SQL statements and click **Execute** to calculate the 10 products users like most from the original data table in the OBS bucket and save the result to the **top_like_product** table.

```
INSERT
 OVERWRITE table top_like_product
```

```
SELECT
  product.brand as brand,
  COUNT(product.brand) as like_count
FROM
  action
  JOIN product ON (action.product_id = product.product_id)
WHERE
  action.type = 'like'
group by
  brand
ORDER BY
  like_count desc
LIMIT
  10
```

Figure 2-10 Script for analyzing the 10 products users like most



* Connection: dli
Database: bi
* Queue: default

```
-- DLI sql
-- ******************************************************************** --
-- author: ei_dlf_100341563
-- create time: 2022/Mar/07 15:04:17 GMT+08:00
-- ******************************************************************** --
INSERT
  OVERWRITE table top_like_product
SELECT
  product.brand as brand,
  COUNT(product.brand) as like_count
FROM
  action
  JOIN product ON (action.product_id = product.product_id)
WHERE
  action.type = 'like'
group by
  brand
ORDER BY
  like_count desc
LIMIT
  10
```

The key parameters are as follows:

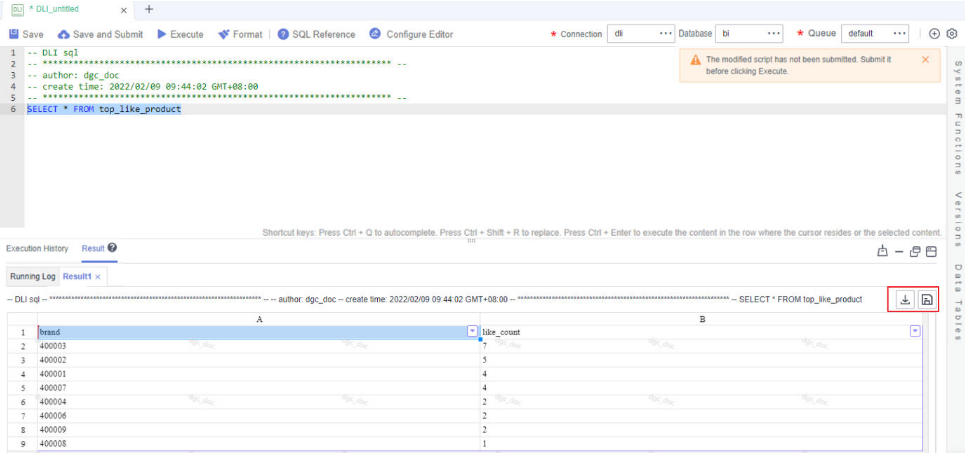
- **Data Connection**: DLI data connection created in **Step 3**
- **Database**: database created in **step 5**
- **Resource Queue**: the default resource queue **default**

Step 4 After debugging the script, click **Save** to save the script and name it **top_like_product**. Click **Submit** to submit the script version. This script will be referenced later in **Developing and Scheduling a Job**.

Step 5 After the script is saved and executed successfully, you can use the following SQL statement to view data in the **top_like_product** table. You can also download or dump the table data by referring to **Figure 2-11**.

```
SELECT * FROM top_like_product
```

Figure 2-11 Viewing the data in the top_like_product table



```
-- DLI sql
-- ******************************************************************** --
-- author: dgc_doc
-- create time: 2022/02/09 09:44:02 GMT+08:00
-- ******************************************************************** --
SELECT * FROM top_like_product
```

The modified script has not been submitted. Submit it before clicking Execute.

Shortcut keys: Press Ctrl + Q to autocomplete. Press Ctrl + Shift + R to replace. Press Ctrl + Enter to execute the content in the row where the cursor resides or the selected content.

-- DLI sql -- ** -- -- author: dgc_doc -- create time: 2022/02/09 09:44:02 GMT+08:00 -- ** -- SELECT * FROM top_like_product

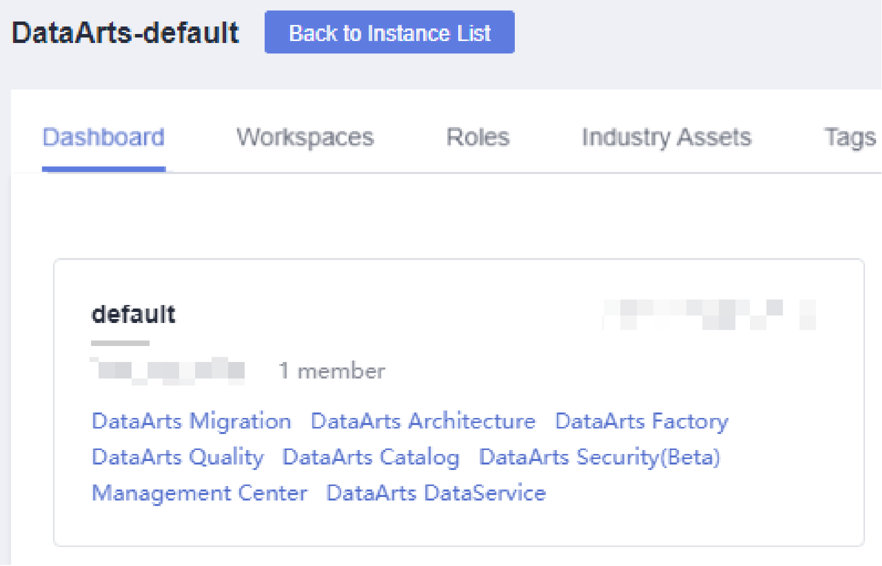
	A	B
1	brand	like_count
2	400003	7
3	400002	5
4	400001	4
5	400007	4
6	400004	2
7	400006	2
8	400009	2
9	400008	1

----End

Analyze 10 Products Users Dislike Most

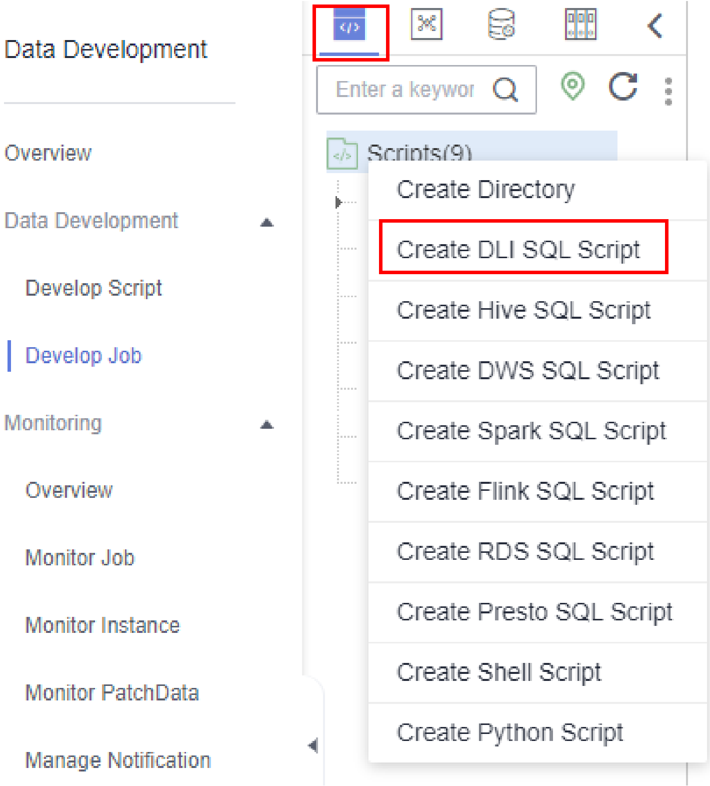
Step 1 Log in to the DataArts Studio console. Locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Factory**.

Figure 2-12 DataArts Factory



Step 2 Create a DLI SQL script used to create data tables by entering DLI SQL statements in the editor.

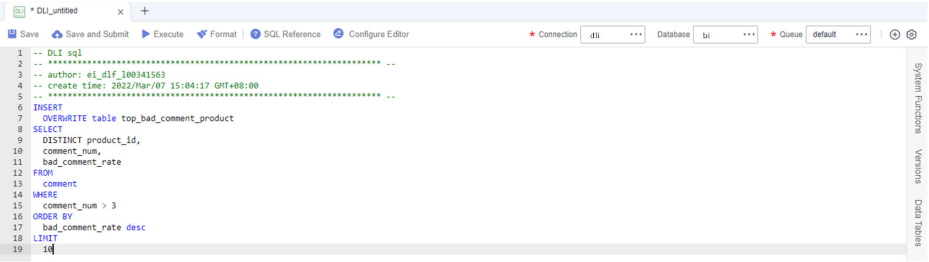
Figure 2-13 Creating a script



Step 3 In the SQL editor, enter the following SQL statements and click **Execute** to calculate the 10 products users dislike most from the original data table in the OBS bucket and save the result to the **top_bad_comment_product** table.

```
INSERT
  OVERWRITE table top_bad_comment_product
SELECT
  DISTINCT product_id,
  comment_num,
  bad_comment_rate
FROM
  comment
WHERE
  comment_num > 3
ORDER BY
  bad_comment_rate desc
LIMIT
  10
```

Figure 2-14 Script for analyzing the 10 products users dislike most



Connection dli
Database bi
Queue default

```
-- DLI sql
-- ************************************************************ --
-- author: ei_dlf_l00341563
-- create time: 2022/Mar/07 15:04:17 GMT+08:00
-- ************************************************************ --
INSERT
  OVERWRITE table top_bad_comment_product
SELECT
  DISTINCT product_id,
  comment_num,
  bad_comment_rate
FROM
  comment
WHERE
  comment_num > 3
ORDER BY
  bad_comment_rate desc
LIMIT
  10
```

The key parameters are as follows:

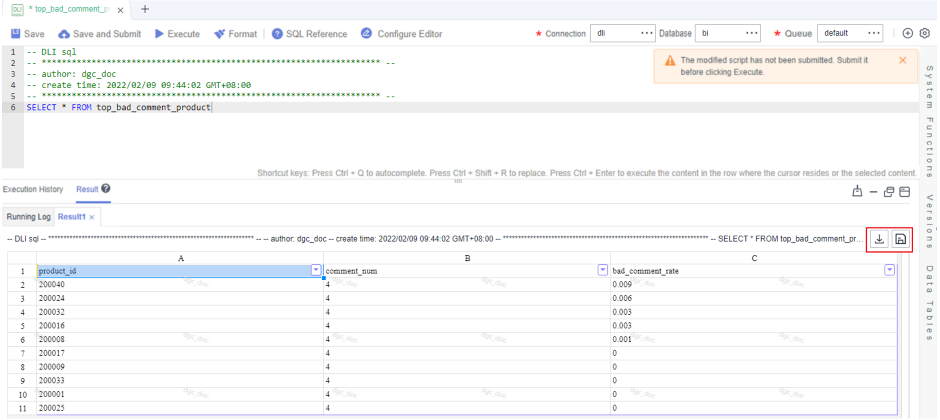
- **Data Connection**: DLI data connection created in **Step 3**
- **Database**: database created in **step 5**
- **Resource Queue**: the default resource queue **default**

Step 4 After debugging the script, click **Save and Submit** to save the script and name it **top_bad_comment_product**. This script will be referenced later in **Developing and Scheduling a Job**.

Step 5 After the script is saved and executed successfully, you can use the following SQL statement to view data in the **top_bad_comment_product** table. You can also download or dump the table data by referring to **Figure 2-15**.

```
SELECT * FROM top_bad_comment_product
```

Figure 2-15 Viewing the data in the top_bad_comment_product table



Connection: dli
Database: bi
Queue: default

```
-- DLI sql
-- ********************************************************************** --
-- author: dgc_doc
-- create time: 2022/02/09 09:44:02 GMT+08:00
-- ********************************************************************** --
SELECT * FROM top_bad_comment_product
```

The modified script has not been submitted. Submit it before clicking Execute.

Shortcut keys: Press Ctrl + Q to autocomplete. Press Ctrl + Shift + R to replace. Press Ctrl + Enter to execute the content in the row where the cursor resides or the selected content.

-- DLI sql -- ** -- -- author: dgc_doc -- create time: 2022/02/09 09:44:02 GMT+08:00 -- ** -- SELECT * FROM top_bad_comment_pr...

	A	B	C
1	product_id	comment_num	bad_comment_rate
2	200040	4	0.009
3	200024	4	0.006
4	200032	4	0.003
5	200016	4	0.003
6	200008	4	0.001
7	200017	4	0
8	200009	4	0
9	200033	4	0
10	200001	4	0
11	200025	4	0

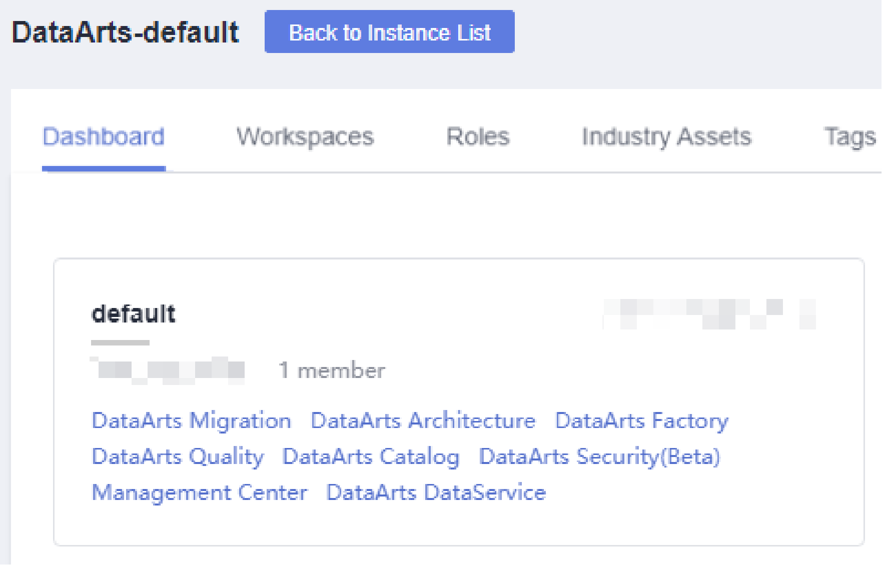
----End

Developing and Scheduling a Job

Assume that the BI reports in the OBS bucket are changing every day. To update the analysis result every day, use the job orchestration and scheduling functions of DataArts Factory.

Step 1 Log in to the DataArts Studio console. Locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Factory**.

Figure 2-16 DataArts Factory



Step 2 Create a batch job named **BI_analysis**.

Figure 2-17 Creating a job

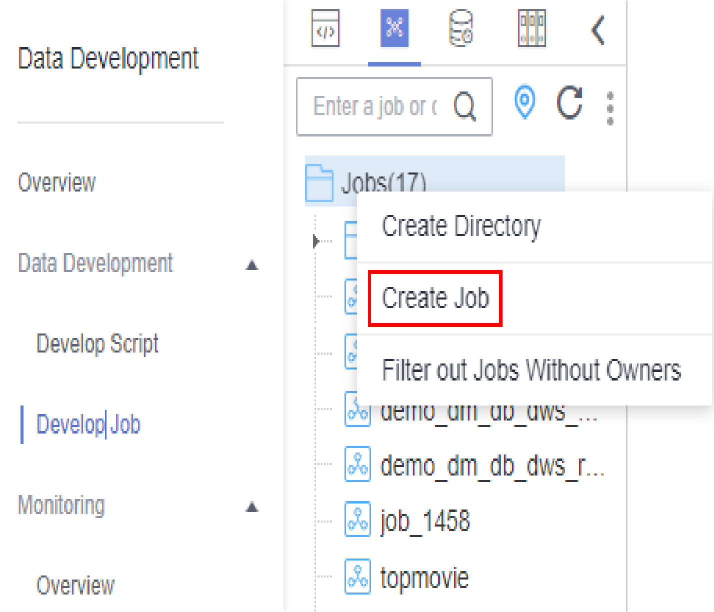
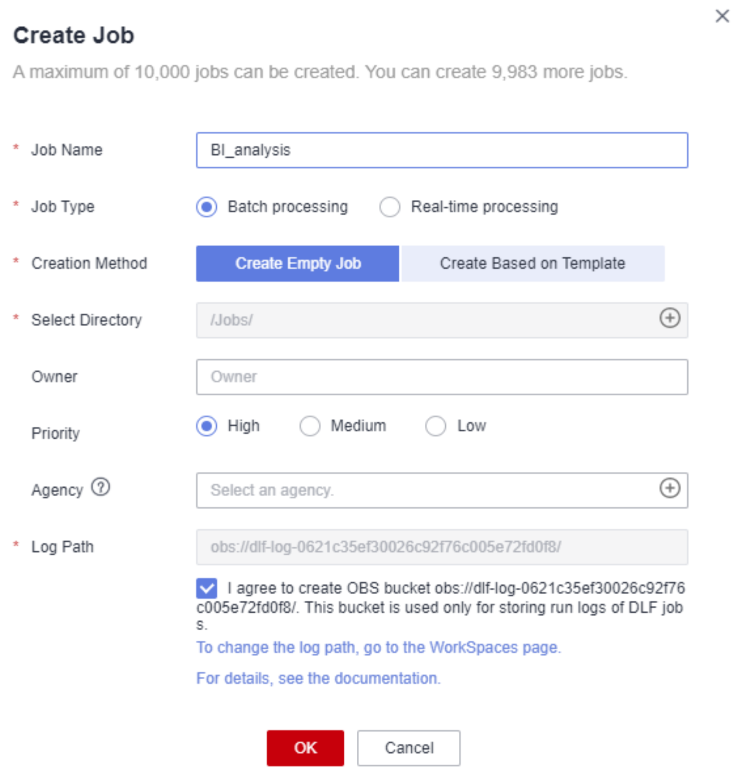


Figure 2-18 Configuring the job



Create Job

A maximum of 10,000 jobs can be created. You can create 9,983 more jobs.

- * Job Name: BI_analysis
- * Job Type: ● Batch processing ○ Real-time processing
- * Creation Method: Create Empty Job | Create Based on Template
- * Select Directory: /Jobs/
- Owner: Owner
- Priority: ● High ○ Medium ○ Low
- Agency ⓘ: Select an agency.
- * Log Path: obs://dlf-log-0621c35ef30026c92f76c005e72fd0f8/

☑ I agree to create OBS bucket obs://dlf-log-0621c35ef30026c92f76c005e72fd0f8/. This bucket is used only for storing run logs of DLF jobs.

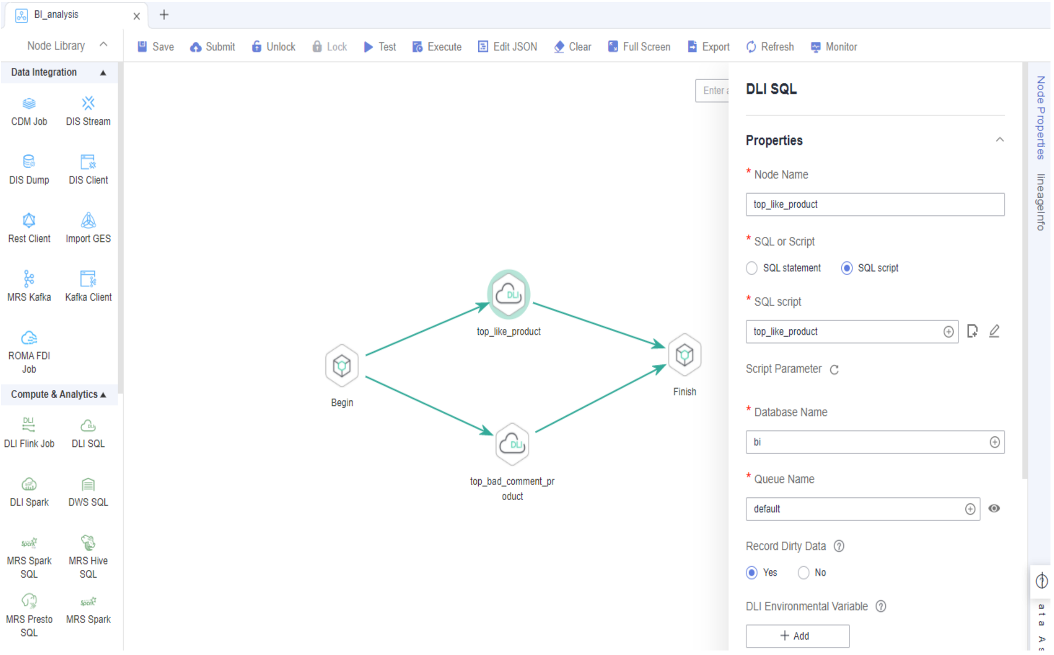
To change the log path, go to the WorkSpaces page.

For details, see the documentation.

OK | Cancel

Step 3 Open the created job, drag two Dummy nodes and two DLI SQL nodes to the canvas, select and drag ⊕, and orchestrate the job shown in **Figure 2-19**.

Figure 2-19 Connecting nodes and configuring node properties



DLI SQL

Properties

- * Node Name: top_like_product
- * SQL or Script: ○ SQL statement ● SQL script
- * SQL script: top_like_product
- Script Parameter
- * Database Name: bi
- * Queue Name: default
- Record Dirty Data ⓘ: ● Yes ○ No
- DLI Environmental Variable ⓘ: + Add

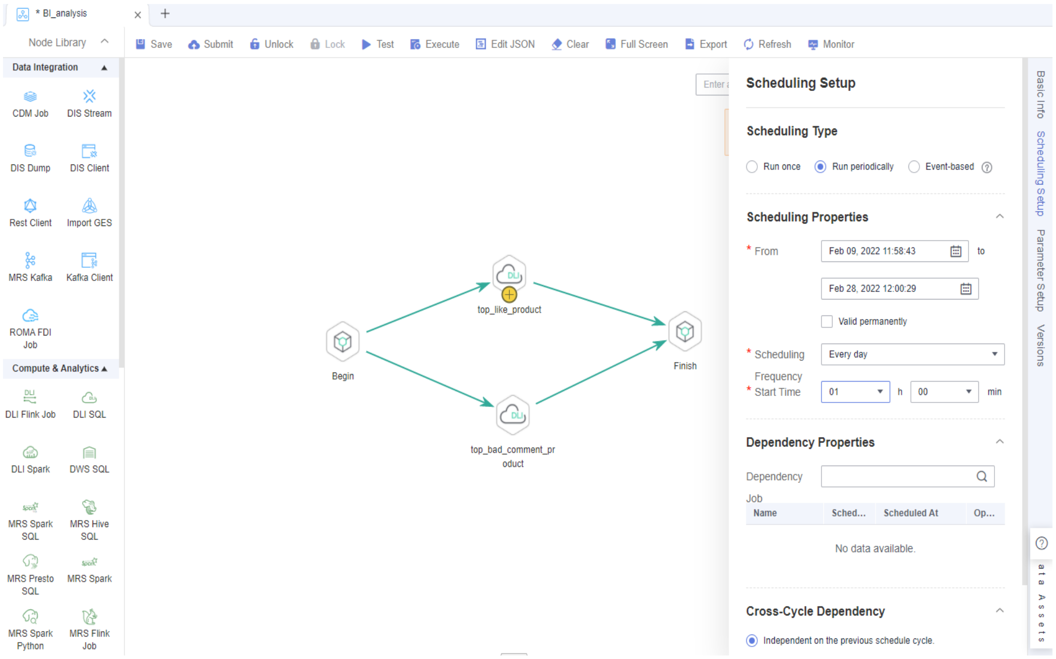
Key nodes:

- **Begin** (Dummy node): serves only as a start identifier.
- **top_like_product** (DLI SQL node): In **Node Properties**, associates with the DLI SQL script **top_like_product** developed in **Analyze 10 Products Users Like Most**.
- **top_bad_comment_product** (DLI SQL node): In **Node Properties**, associates with the DLI SQL script **top_bad_comment_product** developed in **Analyze 10 Products Users Dislike Most**.
- **Finish** (Dummy node): serves only as an end identifier.

Step 4 Click ▶ to test the job.

Step 5 If the job runs properly, click **Scheduling Setup** in the right pane and configure the scheduling policy for the job.

Figure 2-20 Configuring scheduling



Scheduling Setup

Scheduling Type

○ Run once ● Run periodically ○ Event-based

Scheduling Properties

* From: Feb 09, 2022 11:58:43 to Feb 28, 2022 12:00:29

□ Valid permanently

* Scheduling Frequency: Every day

* Start Time: 01 h 00 min

Dependency Properties

Dependency Job:

Name	Sched...	Scheduled At	Op...

No data available.

Cross-Cycle Dependency

● Independent on the previous schedule cycle.

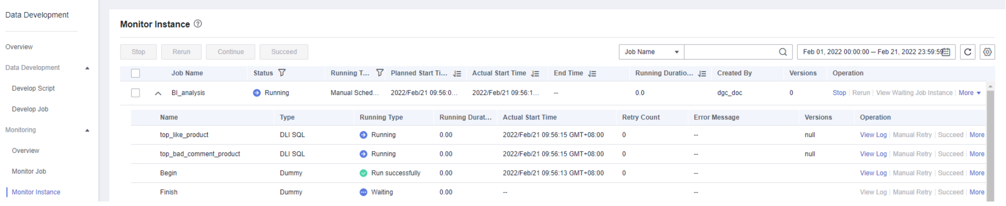
Note:

- **Scheduling Type**: Select **Run periodically**.
- **Scheduling Properties**: The job is executed at 01:00 every day from Feb 09 to Feb 28, 2022.
- **Dependency Properties**: You can configure a dependency job for this job. You do not need to configure it in this practice.
- **Cross-Cycle Dependency**: Select **Independent on the previous schedule cycle**.

Step 6 Click **Save and Submit** and **Execute**. Then the job will be automatically executed every day and the BI report analysis result is automatically saved to the **top_like_product** and **top_bad_comment_product** tables, respectively.

Step 7 If you want to check the job execution result, choose **Monitoring** > **Monitor Instance** in the left navigation pane.

Figure 2-21 Viewing the job execution status



	Job Name	Status	Running T...	Planned Start Ti...	Actual Start Time	End Time	Running Duratio...	Created By	Versions	Operation
□	BI_analysis	Running	Manual Sched...	2022/Feb/21 09:56:0...	2022/Feb/21 09:56:1...	--	0.0	dgc_doc	0	Stop | Rerun | View Waiting Job Instance | More

Name	Type	Running Type	Running Durat...	Actual Start Time	Retry Count	Error Message	Versions	Operation
top_like_product	DLI SQL	Running	0.00	2022/Feb/21 09:56:15 GMT+08:00	0	--	null	View Log | Manual Retry | Succeed | More
top_bad_comment_product	DLI SQL	Running	0.00	2022/Feb/21 09:56:15 GMT+08:00	0	--	null	View Log | Manual Retry | Succeed | More
Begin	Dummy	Run successfully	0.00	2022/Feb/21 09:56:13 GMT+08:00	0	--		View Log | Manual Retry | Succeed | More
Finish	Dummy	Waiting	0.00	--		--		View Log | Manual Retry | Succeed | More

----End

You can also configure notifications to be sent through SMS messages or emails, when a job encounters exceptions or fails.

Now you have learned the data development process based on e-commerce BI reports. In addition, you can analyze the age distribution and gender ratio of users and their browsing, purchase, and evaluation of products to provide valuable information for marketing decision-making, advertising, credit rating, brand monitoring, and user behavior prediction.

2.4 (Optional) Step 3: Unsubscribe from Services

In this development scenario, DataArts Studio, OBS, and DLI incur fees. If you configure notifications, you may be billed for the following service:

- SMN: If you enable SMN notifications for your DataArts Studio modules, you need to pay for the notifications.

After the development is complete, unsubscribe from DataArts Studio and other related services and delete resources in a timely manner to avoid undesired fees.

Table 2-5 Unsubscription methods for services

Service	Billing	Unsubscription Method
DataArts Studio	**DataArts Studio Billing**	DataArts Studio instances support only the yearly/monthly billing mode. You can unsubscribe from the package by referring to **Unsubscriptions**.
OBS	**OBS Billing**	OBS supports pay-per-use and yearly/monthly billing modes. Packages cannot be unsubscribed. In this example, the pay-per-use billing mode is used. You can delete the created bucket after using it. In addition, DataArts Studio job logs and DLI dirty data are stored in an OBS bucket named **dlf-log-{***Project id***}** by default. You can delete the bucket after unsubscribing from DataArts Studio.

Service	Billing	Unsubscription Method
DLI	**DLI Billing**	If you do not purchase a dedicated queue, you will be billed for storage and the amount of data scanned. You will be billed for the amount of data scanned when you submit a job using the default queue, and no fee will be incurred if you do not use a queue. To stop yourself from being billed for storage, delete related data on the **Data Management** page on the DLI console.
SMN	**SMN Billing**	You pay only for what you use. After you unsubscribe from DataArts Studio, no notification will be generated. You can also delete the topics and subscriptions that have been generated.

3 Beginners: Data Integration and Development Based on Movie Scores

3.1 Scenario

In this practice, you will learn how to use Cloud Data Migration (CDM), DataArts Factory of DataArts Studio, and GaussDB(DWS) to analyze movie scores and find out the 10 best and most frequently scored movies. You will learn the data migration function of DataArts Migration, and the script development, job development, and job scheduling functions of DataArts Factory, as well as basic SQL syntax of GaussDB(DWS).

📖 NOTE

This practice involves the DataArts Migration, Management Center, and DataArts Factory modules of DataArts Studio. All DataArts Studio versions can meet requirements.

The procedure is as follows:

1. Make preparations, including **Preparations**, **preparing data sources**, **preparing a data lake**, and **preparing authentication data**.
2. Create a job to migrate data from OBS to DWS. For details, see **Migrating Data from OBS to DWS**.
3. Develop data, including creating DWS SQL scripts and a job.
   - **Creating DWS SQL Script top_rating_movie for Storing 10 Top-rated Movies**
   - **Creating DWS SQL Script top_active_movie for Storing 10 Most Frequently Scored Movies**
   - **Developing and Scheduling a Job**. After orchestrating the job and configuring scheduling policies to periodically execute the job, you can obtain the latest top 10 movies every day.
4. **Unsubscribe from services**. If you do not want to use DataArts Studio and related services, unsubscribe from them and delete resources in a timely manner.

3.2 Step 1: Make Preparations

Preparations

If you use DataArts Studio for the first time, register a , buy a DataArts Studio instance, and create a workspace by following the instructions provided in **Preparations**. Then you can go to the created workspace and start using DataArts Studio.

Preparing Data Sources

This practice uses the 100,000 scores given by 1,000 users to 1,700 movies. The scores are available at **https://grouplens.org/datasets/movielens/100k/**. Obtain the .zip package from the link and extract the **u.item** and **u.data** files from it. They contain the movie information and rating information, respectively.

To facilitate demonstration, this practice provides some data used to simulate the original data. To integrate the source data into the cloud, you need to store the sample data in CSV files and upload them to an OBS bucket.

Step 1 Create CSV files (UTF-8 without BOM), name the files with the corresponding data table names, copy the sample data to different CSV files, and save the files.

To generate a CSV file in Windows, you can perform the following steps:

1. Use a text editor (for example, Notepad) to create a .txt document and copy the sample data to the document. Then check the total number of rows and check whether the data of rows is correctly separated. (If the sample data is copied from a PDF document, the data in a single row will be wrapped if the data is too long. In this case, you must manually adjust the data to ensure that it is in a single row.)
2. Choose **File** > **Save as**. In the displayed dialog box, set **Save as type** to **All files (*.*)**, enter the file name with the .csv suffix for **File name**, and select the UTF-8 encoding format (without BOM) to save the file in CSV format.

Step 2 Upload the CSV file to OBS.

1. Log in to the management console and choose **Storage** > **Object Storage Service** to access the OBS console.
2. Click **Create Bucket** and set parameters as prompted to create an OBS bucket named **fast-demo**.

   NOTE

   To ensure network connectivity, select the same region for OBS bucket as that for the DataArts Studio instance. If an enterprise project is required, select the enterprise project that is the same as that of the DataArts Studio instance.

   For details about how to create a bucket on the OBS console, see **Creating a Bucket** in *Object Storage Service Console Operation Guide*.
3. Upload data to OBS bucket **fast-demo**.

   For details about how to upload a file on the OBS console, see **Uploading a File** in *Object Storage Service Console Operation Guide*.

----End

This practice involves movie data (**movies.csv**) and rating data (**ratings.csv**). Descriptions of the data are as follows:

- **movies.csv**:

```
movieId,movieTitle,videoReleaseDate,IMDbURL,unknown,Action,Adventure,Animation,Childrens,Comed
y,Crime,Documentary,Drama,Fantasy,FilmNoir,Horror,Musical,Mystery,Romance,SciFi,Thriller,War,Wester
n
1,Toy Story (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Toy%20Story
%20(1995),0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0
2,GoldenEye (1995),1-Jan-95,http://us.imdb.com/M/title-exact?GoldenEye
%20(1995),0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0
3,Four Rooms (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Four%20Rooms
%20(1995),0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0
4,Get Shorty (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Get%20Shorty
%20(1995),0,1,0,0,0,1,0,0,1,0,0,0,0,0,0,0,0,0,0
5,Copycat (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Copycat
%20(1995),0,0,0,0,0,0,1,0,1,0,0,0,0,0,0,0,1,0,0
6,Shanghai Triad (Yao a yao yao dao waipo qiao) (1995),1-Jan-95,http://us.imdb.com/Title?Yao+a+yao
+yao+dao+waipo+qiao+(1995),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0
7,Twelve Monkeys (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Twelve%20Monkeys
%20(1995),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,1,0,0,0
8,Babe (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Babe
%20(1995),0,0,0,0,1,1,0,0,1,0,0,0,0,0,0,0,0,0,0
9,Dead Man Walking (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Dead%20Man%20Walking
%20(1995),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0
10,Richard III (1995),22-Jan-96,http://us.imdb.com/M/title-exact?Richard%20III
%20(1995),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,1,0
11,Seven (Se7en) (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Se7en
%20(1995),0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,1,0,0
12,"Usual Suspects, The (1995)",14-Aug-95,"http://us.imdb.com/M/title-exact?Usual
%20Suspects,%20The%20(1995)",0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,1,0,0
13,Mighty Aphrodite (1995),30-Oct-95,http://us.imdb.com/M/title-exact?Mighty%20Aphrodite
%20(1995),0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0
14,"Postino, Il (1994)",1-Jan-94,"http://us.imdb.com/M/title-exact?Postino,%20Il
%20(1994)",0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0
15,Mr. Holland's Opus (1995),29-Jan-96,http://us.imdb.com/M/title-exact?Mr.%20Holland's%20Opus
%20(1995),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0
16,French Twist (Gazon maudit) (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Gazon%20maudit
%20(1995),0,0,0,0,0,1,0,0,0,0,0,0,0,0,1,0,0,0,0
17,From Dusk Till Dawn (1996),5-Feb-96,http://us.imdb.com/M/title-exact?From%20Dusk%20Till
%20Dawn%20(1996),0,1,0,0,0,1,1,0,0,0,0,1,0,0,0,0,1,0,0
18,"White Balloon, The (1995)",1-Jan-95,http://us.imdb.com/M/title-exact?Badkonake%20Sefid
%20(1995),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0
19,Antonia's Line (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Antonia
%20(1995),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0
20,Angels and Insects (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Angels%20and%20Insects
%20(1995),0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0
21,Muppet Treasure Island (1996),16-Feb-96,http://us.imdb.com/M/title-exact?Muppet%20Treasure
%20Island%20(1996),0,1,1,0,0,1,0,0,0,0,0,0,1,0,0,0,1,0,0
22,Braveheart (1995),16-Feb-96,http://us.imdb.com/M/title-exact?Braveheart
%20(1995),0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,1,0
23,Taxi Driver (1976),16-Feb-96,http://us.imdb.com/M/title-exact?Taxi%20Driver
%20(1976),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0
24,Rumble in the Bronx (1995),23-Feb-96,http://us.imdb.com/M/title-exact?Hong%20Faan%20Kui
%20(1995),0,1,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0
25,"Birdcage, The (1996)",8-Mar-96,"http://us.imdb.com/M/title-exact?Birdcage,%20The
%20(1996)",0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0
26,"Brothers McMullen, The (1995)",1-Jan-95,"http://us.imdb.com/M/title-exact?Brothers
%20McMullen,%20The%20(1995)",0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0
27,Bad Boys (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Bad%20Boys
%20(1995),0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0
28,Apollo 13 (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Apollo
%2013%20(1995),0,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0
29,Batman Forever (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Batman%20Forever
%20(1995),0,1,1,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0
30,Belle de jour (1967),1-Jan-67,http://us.imdb.com/M/title-exact?Belle%20de%20jour
%20(1967),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0
31,Crimson Tide (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Crimson%20Tide
%20(1995),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,1,0
```

```
32,Crumb (1994),1-Jan-94,http://us.imdb.com/M/title-exact?Crumb
%20(1994),0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0
33,Desperado (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Desperado
%20(1995),0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0,0
34,"Doom Generation, The (1995)",1-Jan-95,"http://us.imdb.com/M/title-exact?Doom
%20Generation,%20The%20(1995)",0,0,0,0,0,1,0,0,1,0,0,0,0,0,0,0,0,0,0
35,Free Willy 2: The Adventure Home (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Free%20Willy
%202:%20The%20Adventure%20Home%20(1995),0,0,1,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0
36,Mad Love (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Mad%20Love
%20(1995),0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0,0,0
37,Nadja (1994),1-Jan-94,http://us.imdb.com/M/title-exact?Nadja
%20(1994),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0
38,"Net, The (1995)",1-Jan-95,"http://us.imdb.com/M/title-exact?Net,%20The
%20(1995)",0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1,0,0
39,Strange Days (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Strange%20Days
%20(1995),0,1,0,0,0,0,1,0,0,0,0,0,0,0,0,1,0,0,0
40,"To Wong Foo, Thanks for Everything! Julie Newmar (1995)",1-Jan-95,"http://us.imdb.com/M/title-
exact?To%20Wong%20Foo,%20Thanks%20for%20Everything!%20Julie%20Newmar
%20(1995)",0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0
41,Billy Madison (1995),1-Jan-95,http://us.imdb.com/M/title-exact?Billy%20Madison
%20(1995),0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0
42,Clerks (1994),1-Jan-94,http://us.imdb.com/M/title-exact?Clerks
%20(1994),0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0
43,Disclosure (1994),1-Jan-94,http://us.imdb.com/M/title-exact?Disclosure
%20(1994),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0
44,Dolores Claiborne (1994),1-Jan-94,http://us.imdb.com/M/title-exact?Dolores%20Claiborne
%20(1994),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,1,0,0
45,Eat Drink Man Woman (1994),1-Jan-94,http://us.imdb.com/M/title-exact?Yinshi%20Nan%20Nu
%20(1994),0,0,0,0,0,1,0,0,1,0,0,0,0,0,0,0,0,0,0
46,Exotica (1994),1-Jan-94,http://us.imdb.com/M/title-exact?Exotica
%20(1994),0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0
47,Ed Wood (1994),1-Jan-94,http://us.imdb.com/M/title-exact?Ed%20Wood
%20(1994),0,0,0,0,0,1,0,0,1,0,0,0,0,0,0,0,0,0,0
48,Hoop Dreams (1994),1-Jan-94,http://us.imdb.com/M/title-exact?Hoop%20Dreams
%20(1994),0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0
49,I.Q. (1994),1-Jan-94,http://us.imdb.com/M/title-exact?
I.Q.%20(1994),0,0,0,0,0,1,0,0,0,0,0,0,0,0,1,0,0,0,0
50,Star Wars (1977),1-Jan-77,http://us.imdb.com/M/title-exact?Star%20Wars
%20(1977),0,1,1,0,0,0,0,0,0,0,0,0,0,0,1,1,0,1,0
```

The following table describes the data.

Table 3-1 Movie data description

Field	Type	Description
movieId	INT	Movie ID
movieTitle	VARCHAR	Movie name
videoReleaseDate	VARCHAR	Release date
IMDbURL	VARCHAR	IMDb link
unknown	INT	Whether the movie type is unknown. If yes, the value is **1**; otherwise, the value is **0**.
Action	INT	Whether the movie type is action. If yes, the value is **1**; otherwise, the value is **0**.
Adventure	INT	Whether the movie type is adventure. If yes, the value is **1**; otherwise, the value is **0**.

Copyright © Huawei Cloud Computing Technologies Co., Ltd.

Field	Type	Description
Animation	INT	Whether the movie type is animation. If yes, the value is **1**; otherwise, the value is **0**.
Childrens	INT	Whether the movie type is children. If yes, the value is **1**; otherwise, the value is **0**.
Comedy	INT	Whether the movie type is comedy. If yes, the value is **1**; otherwise, the value is **0**.
Crime	INT	Whether the movie type is crime. If yes, the value is **1**; otherwise, the value is **0**.
Documentary	INT	Whether the movie type is documentary. If yes, the value is **1**; otherwise, the value is **0**.
Drama	INT	Whether the movie type is drama. If yes, the value is **1**; otherwise, the value is **0**.
Fantasy	INT	Whether the movie type is fantasy. If yes, the value is **1**; otherwise, the value is **0**.
FilmNoir	INT	Whether the movie type is noir. If yes, the value is **1**; otherwise, the value is **0**.
Horror	INT	Whether the movie type is horror. If yes, the value is **1**; otherwise, the value is **0**.
Musical	INT	Whether the movie type is musical. If yes, the value is **1**; otherwise, the value is **0**.
Mystery	INT	Whether the movie type is mystery. If yes, the value is **1**; otherwise, the value is **0**.
Romance	INT	Whether the movie type is romance. If yes, the value is **1**; otherwise, the value is **0**.
SciFi	INT	Whether the movie type is science fiction. If yes, the value is **1**; otherwise, the value is **0**.
Thriller	INT	Whether the movie type is thriller. If yes, the value is **1**; otherwise, the value is **0**.
War	INT	Whether the movie type is war. If yes, the value is **1**; otherwise, the value is **0**.
Western	INT	Whether the movie type is western. If yes, the value is **1**; otherwise, the value is **0**.

- **ratings.csv**:

```
userId,movieId,rating,timestamp
210,40,3,891035994
224,29,3,888104457
308,1,4,887736532
7,32,4,891350932
10,16,4,877888877
99,4,5,886519097
115,20,3,881171009
138,26,5,879024232
243,15,3,879987440
293,5,3,888906576
162,25,4,877635573
135,23,4,879857765
62,21,3,879373460
59,23,5,888205300
43,14,2,883955745
19,4,4,885412840
5,2,3,875636053
72,48,4,880036718
224,26,3,888104153
299,14,4,877877775
151,10,5,879524921
6,14,5,883599249
250,7,4,878089716
268,2,2,875744173
292,11,5,881104093
181,3,2,878963441
145,15,2,875270655
1,33,4,878542699
276,2,4,874792436
18,26,4,880129731
87,40,3,879876917
272,12,5,879455254
296,20,5,884196921
5,17,4,875636198
128,15,4,879968827
287,1,5,875334088
65,47,2,879216672
1,20,4,887431883
290,50,5,880473582
45,25,4,881014015
109,8,3,880572642
157,25,3,886890787
301,33,4,882078228
62,12,4,879373613
276,40,3,874791871
269,22,1,891448072
10,7,4,877892210
244,17,2,880607205
222,26,3,878183043
185,23,4,883524249
207,13,3,875506839
8,22,5,879362183
222,49,3,878183512
200,11,5,884129542
90,25,5,891384789
15,25,3,879456204
234,10,3,891227851
295,39,4,879518279
217,2,3,889069782
189,20,5,893264466
42,44,3,881108548
268,21,3,875742822
262,28,3,879792220
90,22,4,891384357
270,25,5,876954456
194,23,4,879522819
161,48,1,891170745
```

```
58,9,4,884304328
79,50,4,891271545
221,48,5,875245462
223,11,3,891550649
292,9,4,881104148
16,8,5,877722736
17,13,3,885272654
148,1,4,877019411
280,1,4,891700426
110,38,3,886988574
90,12,5,891383241
239,9,5,889180446
311,9,4,884963365
151,13,3,879542688
2,50,5,888552084
8,50,5,879362124
286,44,3,877532173
85,25,2,879452769
274,50,5,878944679
217,27,1,889070011
181,14,1,878962392
297,25,4,874954497
1,47,4,875072125
6,23,4,883601365
222,22,5,878183285
314,28,5,877888346
291,15,5,874833668
94,24,4,885873423
83,43,4,880308690
43,40,3,883956468
44,15,4,878341343
158,24,4,880134261
151,12,5,879524368
66,1,3,883601324
5,1,4,875635748
207,25,4,876079113
109,1,4,880563619
227,50,4,879035347
181,1,3,878962392
213,13,4,878955139
121,14,5,891390014
117,15,5,880125887
85,13,3,879452866
313,22,3,891014870
43,5,4,875981421
11,38,3,891905936
72,28,4,880036824
115,8,5,881171982
95,1,5,879197329
145,22,5,875273021
66,7,3,883601355
267,17,4,878971773
25,25,5,885853415
103,24,4,880415847
87,9,4,879877931
49,47,5,888068715
135,39,3,879857931
269,13,4,891446662
99,50,5,885679998
306,14,5,876503995
291,7,5,874834481
312,28,4,891698300
184,36,3,889910195
305,11,1,886323237
198,7,4,884205317
104,7,3,888465972
293,39,3,888906804
256,25,5,882150552
92,15,3,875640189
```

```
1,17,3,875073198
214,42,5,892668130
82,14,4,876311280
305,50,5,886321799
223,8,2,891550684
91,28,4,891439243
315,13,4,879821158
269,9,4,891446246
217,7,4,889069741
49,7,4,888067307
87,2,4,879876074
268,1,3,875742341
262,47,2,879794599
84,12,5,883452874
264,33,3,886122644
224,20,1,888104487
200,24,2,884127370
92,24,3,875640448
276,38,3,874792574
286,34,5,877534701
49,38,1,888068289
311,5,3,884365853
269,47,4,891448386
194,4,4,879521397
57,28,4,883698324
108,50,4,879879739
207,4,4,876198457
181,16,1,878962996
94,9,5,885872684
234,20,4,891227979
68,7,3,876974096
13,14,4,884538727
98,47,4,880498898
53,24,3,879442538
239,10,5,889180338
63,20,3,875748004
276,43,1,874791383
272,48,4,879455143
116,7,2,876453915
26,25,3,891373727
62,24,4,879372633
295,47,5,879518166
63,50,4,875747292
49,17,2,888068651
310,24,4,879436242
7,44,5,891351728
326,22,4,879874989
213,12,5,878955409
222,29,3,878184571
249,11,5,879640868
217,22,5,889069741
189,1,5,893264174
234,50,4,892079237
296,48,5,884197091
81,3,4,876592546
151,15,4,879524879
59,12,5,888204260
246,8,3,884921245
276,34,2,877934264
97,50,5,884239471
244,7,4,880602558
298,8,5,884182748
7,28,5,891352341
41,28,4,890687353
```

The following table describes the data.

Table 3-2 Rating data description

Field	Type	Description
userId	INT	User ID
movieId	INT	Movie ID
rating	INT	Rating. The total score is 5 points.
timestamp	VARCHAR	Timestamp

Preparing a Data Lake

This practice uses DWS as the data lake.

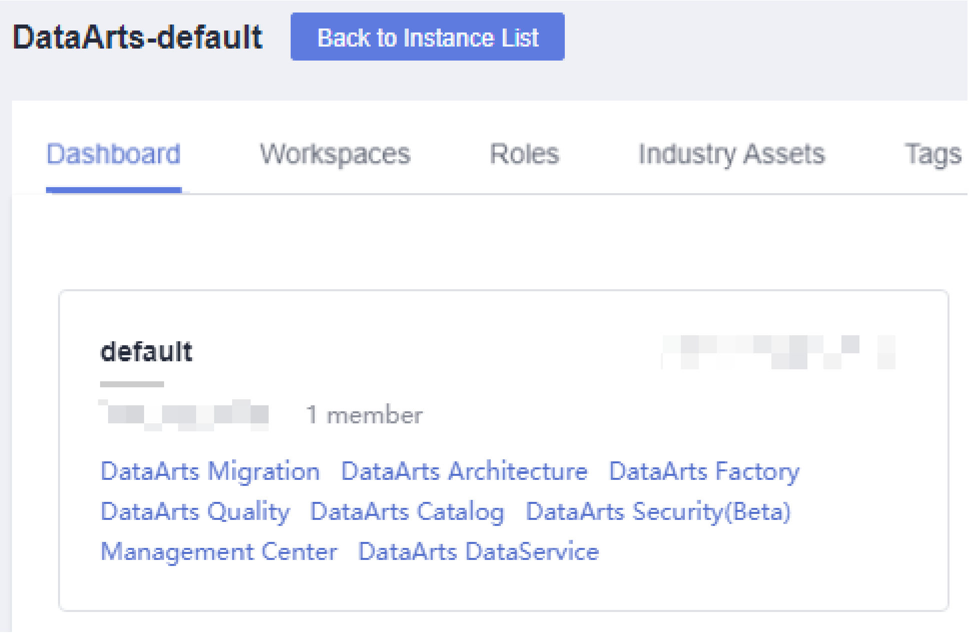
For details about how to create a DWS cluster, see **Creating a Cluster**. The DWS cluster must meet the following requirements so that it can communicate with the DataArts Studio instance:

- If the CDM cluster in the DataArts Studio instance and the DWS cluster are in different regions, a public network or a dedicated connection is required.
- If the CDM cluster in the DataArts Studio instance and the DWS cluster 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 DWS 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.

After creating a DWS cluster, you need to create a DWS connection in Management Center, create a database and schema through the DataArts Factory module, and run an SQL statement to create a DWS table. The procedure is as follows:

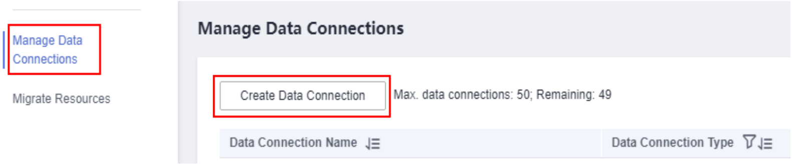
Step 1 On the DataArts Studio console, locate a workspace and click **Management Center**.

Figure 3-1 Management Center



Step 2 On the displayed **Manage Data Connections** page, click **Create Data Connection**.

Figure 3-2 Creating a data connection

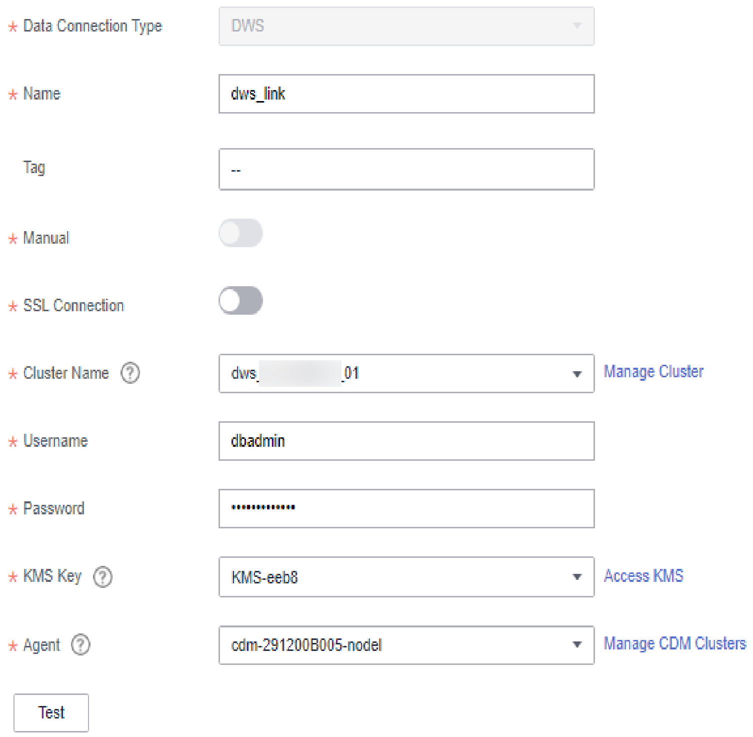


Max. data connections: 50; Remaining: 49

Step 3 On the displayed page, configure the following parameters and click **OK**. **Figure 3-3** lists the parameters.

- **Data Connection Type**: Select **DWS**.
- **Name**: Enter **dws_link**.
- **Manual**: Disable it, and you do not need to manually specify **IP Address** and **Port**.
- **Cluster Name**: Select the DWS cluster that you have created.
- **Username**: Enter the database username that you specified when creating the DWS cluster. The default username is **dbadmin**.
- **Password**: Enter the password that you specified when creating the DWS cluster for accessing the database.
- **KMS Key**: Select a KMS key used to encrypt sensitive data. If no KMS key is available, click **Access KMS** to go to the KMS console and create one.
- **Agent**: Select a CDM cluster as the connection agent. The CDM cluster must be able to communicate with the DWS cluster. In this example, you can select the DataArts Migration cluster that is automatically created when the DataArts Studio instance is created.

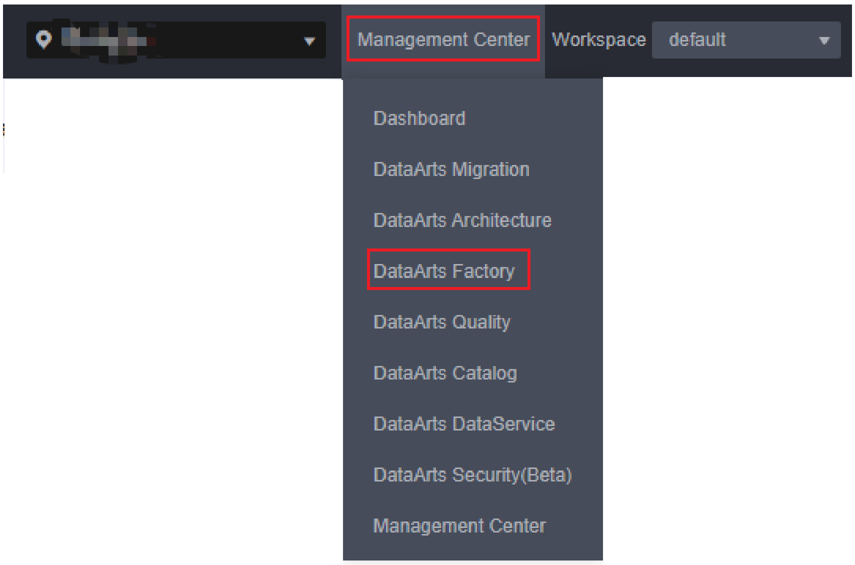
Figure 3-3 DWS connection parameters



* Data Connection Type	DWS
* Name	dws_link
Tag	--
* Manual	
* SSL Connection	
* Cluster Name	dws_[redacted]01
* Username	dbadmin
* Password	•••••••••••••
* KMS Key	KMS-eeb8
* Agent	cdm-291200B005-nodel

Step 4 Go to the **DataArts Factory** page.

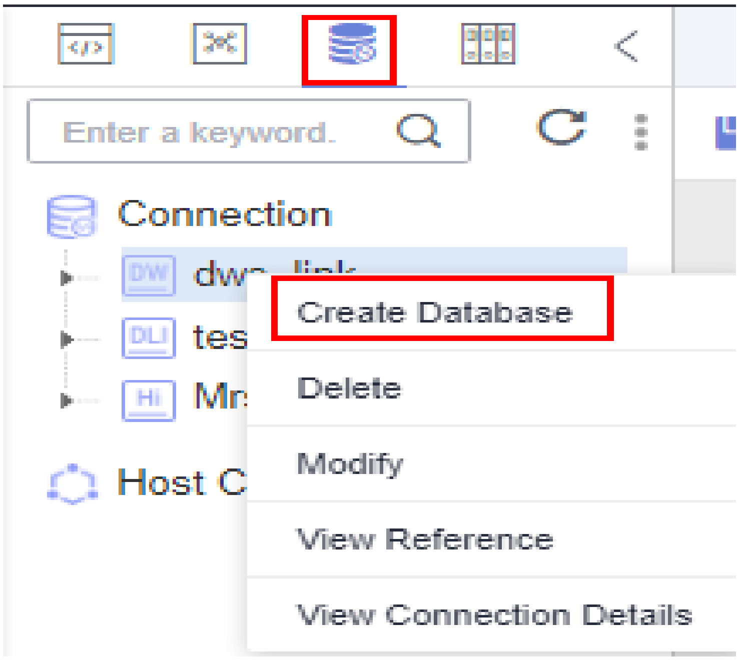
Figure 3-4 DataArts Factory page



Step 5 Create a DWS database and a database schema.

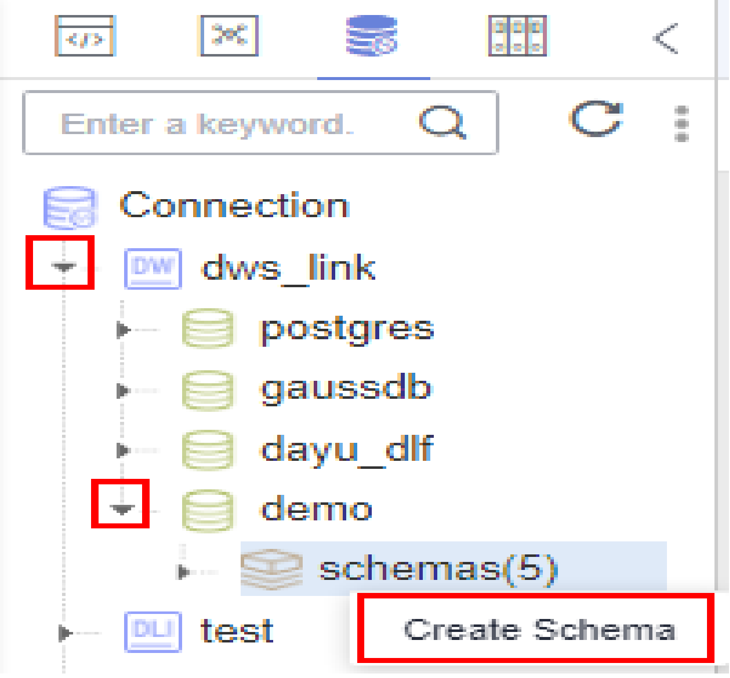
1. Right-click the DWS connection and select **Create Database** to create a database named **demo** for storing data tables.

Figure 3-5 Creating a database



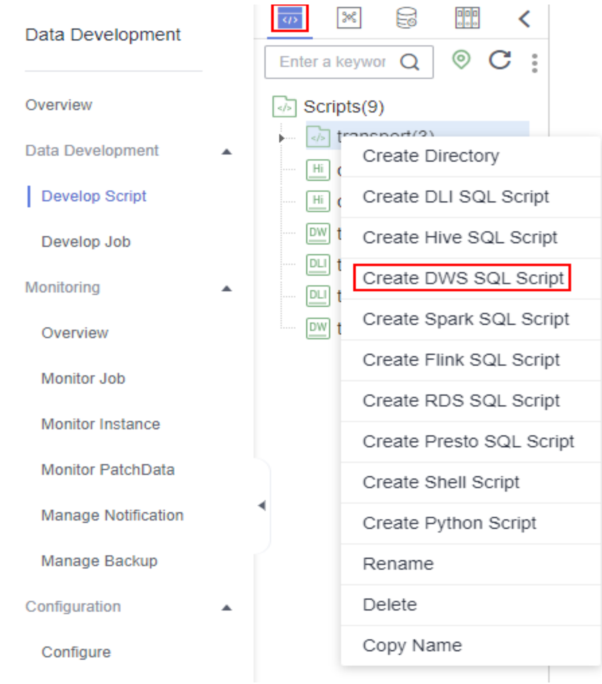
2. Expand the DWS connection directory to the database schema level, right-click **schemas**, and select **Create Schema** to create a schema named **dgc** for storing data tables.

Figure 3-6 Creating a database schema



Step 6 Create a DWS SQL script used to create data tables by entering DWS SQL statements in the editor.

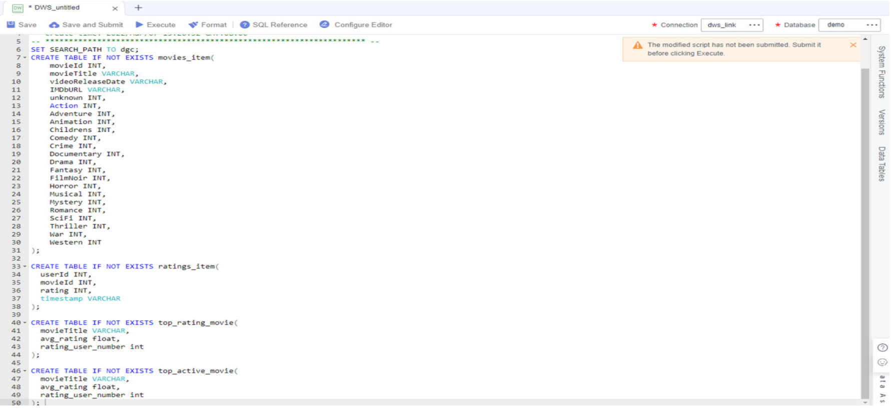
Figure 3-7 Creating a script



Step 7 In the SQL editor, enter the following SQL statements and click **Execute** to create data tables. Among them, **movies_item** and **ratings_item** are original data tables, to which data will be migrated from OBS through CDM. **top_rating_movie** and **top_active_movie** are result tables which store analysis results.

```
SET SEARCH_PATH TO dgc;
CREATE TABLE IF NOT EXISTS movies_item(
    movieId INT,
    movieTitle VARCHAR,
    videoReleaseDate VARCHAR,
    IMDbURL VARCHAR,
    unknown INT,
    Action INT,
    Adventure INT,
    Animation INT,
    Childrens INT,
    Comedy INT,
    Crime INT,
    Documentary INT,
    Drama INT,
    Fantasy INT,
    FilmNoir INT,
    Horror INT,
    Musical INT,
    Mystery INT,
    Romance INT,
    SciFi INT,
    Thriller INT,
    War INT,
    Western INT
);

CREATE TABLE IF NOT EXISTS ratings_item(
  userId INT,
  movieId INT,
  rating INT,
  timestamp VARCHAR
);

CREATE TABLE IF NOT EXISTS top_rating_movie(
  movieTitle VARCHAR,
  avg_rating float,
  rating_user_number int
);

CREATE TABLE IF NOT EXISTS top_active_movie(
  movieTitle VARCHAR,
  avg_rating float,
  rating_user_number int
);
```

Figure 3-8 Creating data tables



* Connection dws_link

* Database demo

The modified script has not been submitted. Submit it before clicking Execute.

```
-- ****************************************************************** --
SET SEARCH_PATH TO dgc;
CREATE TABLE IF NOT EXISTS movies_item(
    movieId INT,
    movieTitle VARCHAR,
    videoReleaseDate VARCHAR,
    IMDbURL VARCHAR,
    unknown INT,
    Action INT,
    Adventure INT,
    Animation INT,
    Childrens INT,
    Comedy INT,
    Crime INT,
    Documentary INT,
    Drama INT,
    Fantasy INT,
    FilmNoir INT,
    Horror INT,
    Musical INT,
    Mystery INT,
    Romance INT,
    SciFi INT,
    Thriller INT,
    War INT,
    Western INT
);

CREATE TABLE IF NOT EXISTS ratings_item(
  userId INT,
  movieId INT,
  rating INT,
  timestamp VARCHAR
);

CREATE TABLE IF NOT EXISTS top_rating_movie(
  movieTitle VARCHAR,
  avg_rating float,
  rating_user_number int
);

CREATE TABLE IF NOT EXISTS top_active_movie(
  movieTitle VARCHAR,
  avg_rating float,
  rating_user_number int
);
```

The key parameters are as follows:

- **Data Connection**: DWS data connection created in **Step 3**
- **Database**: database created in **step 5**

Step 8 After the script is executed successfully, run the following script to check whether the data tables are created successfully. After confirming that the data tables are created successfully, you can close the script as it is no longer needed.

```
SELECT * FROM pg_tables;
```

----End

Preparing Authentication Data

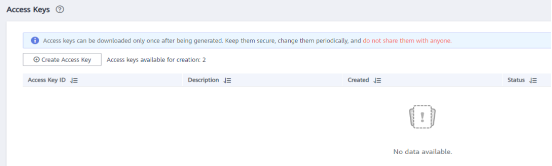
If you want to migrate OBS data using CDM, you need AK/SK authentication. Therefore, you must create an AK/SK pair.

- Access Key ID (AK): indicates the ID of the access key, which is a unique identifier associated with a secret access key and is used in conjunction with a secret access key to sign requests cryptographically.
- Secret Access Key (SK): indicates the key used with its associated AK to cryptographically sign requests and identify request senders to prevent requests from being modified.

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 3-9**.

Figure 3-9 Clicking Create Access Key



Access keys can be downloaded only once after being generated. Keep them secure, change them periodically, and do not share them with anyone.

Access keys available for creation: 2

No data available.

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

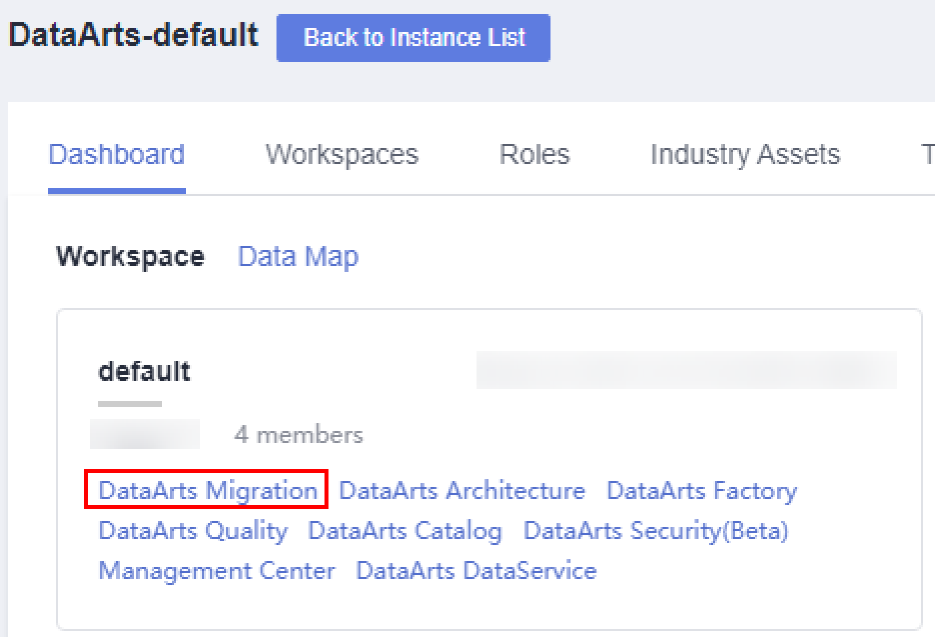
- Only two access keys can be added for each user.
- To ensure access key security, the access key is automatically downloaded only when it is generated for the first time and cannot be obtained from the management console later. Keep them properly.

3.3 Step 2: Integrate Data

Migrating Data from OBS to DWS

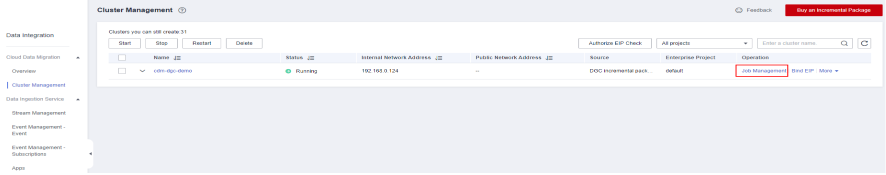
Step 1 Log in to the DataArts Studio console. Locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Migration**.

Figure 3-10 DataArts Migration



Step 2 On the displayed **Cluster Management** page, click **Job Management** in the **Operation** column.

Figure 3-11 Job Management



Clusters you can still create:31

Name	Status	Internal Network Address	Public Network Address	Source	Enterprise Project	Operation
cdm-dgc-demo	Running	192.168.0.124	--	DGC incremental pack...	default	Job Management Bind EIP More

Step 3 Click the **Links** tab and then **Create Link**.

Step 4 On the **Create Link** page, select **Object Storage Service (OBS)** and click **Next** to create a link named **obs_link** from CDM to OBS.

Table 3-3 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
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.	-
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

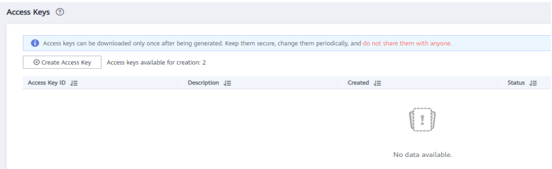
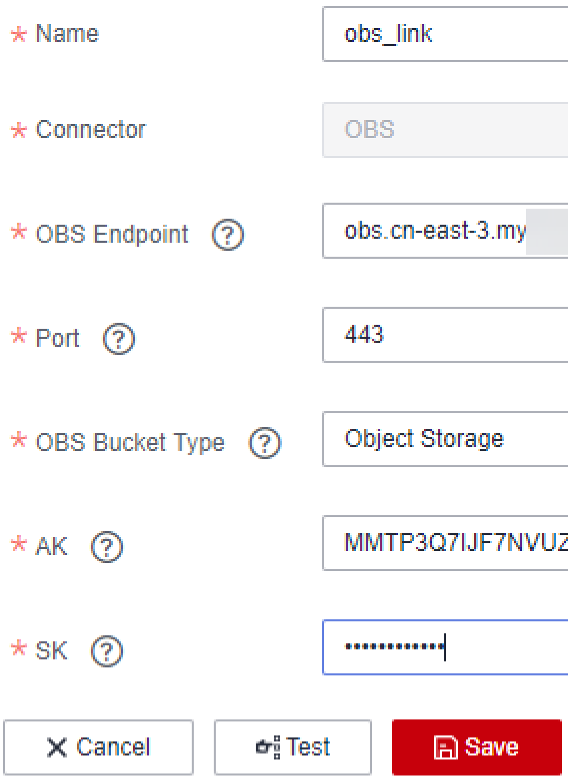
Parameter	Description	Example Value
AK	AK and SK are used to log in to the OBS server. You need to create an access key for the current account and obtain an AK/SK pair. To obtain an access key, perform the following steps: 1. Log in to the management console, move the cursor to the username in the upper 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 3-12**. **Figure 3-12** Clicking Create Access Key Access Keys Access keys can be downloaded only once after being generated. Keep them secure, change them periodically, and do not share them with anyone. Create Access Key Access keys available for creation: 2 Access Key ID Description Created Status No data available. 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		-

Figure 3-13 Creating an OBS link



* Name: obs_link

* Connector: OBS

* OBS Endpoint: obs.cn-east-3.my[redacted].cor

* Port: 443

* OBS Bucket Type: Object Storage

* AK: MMTP3Q7IJF7NVUZA9AOC

* SK: ••••••••••••

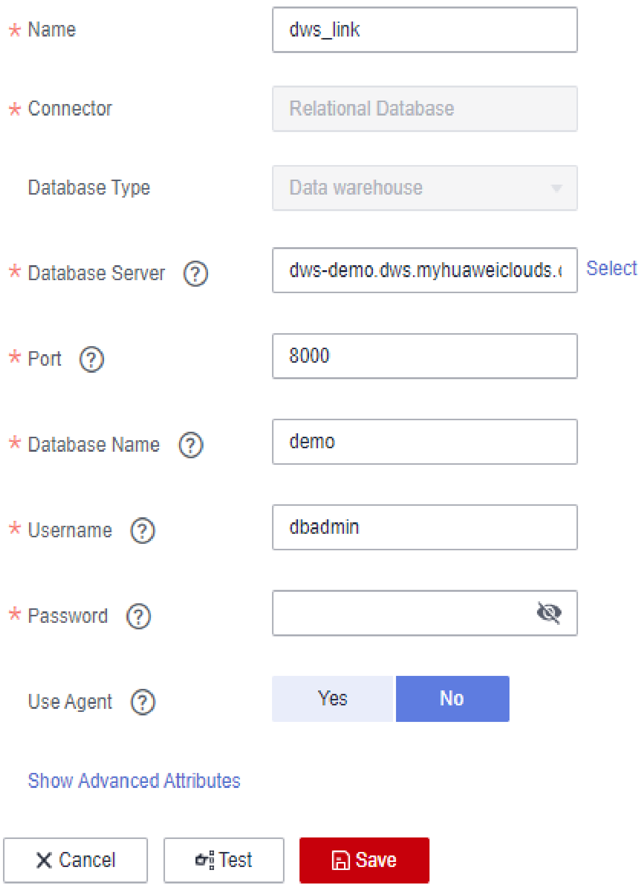
Step 5 On the **Create Link** page, select **Data Warehouse Service** and click **Next** to create a link named **dws_link** from CDM to DWS.

Table 3-4 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
Database Server	Click **Select** next to the text box to obtain the list of DWS instances.	-
Port	Port of the target database. The DWS database port is **8000** by default.	8000
Database Name	Name of the target database	demo
Username	Username used for accessing the database. This account must have the permissions to read and write data tables and metadata.	dbadmin
Password	User password	-

Parameter	Description	Example Value
Use Agent	Whether to extract data from the data source through an agent	No

Figure 3-14 Creating a DWS link



* Name: dws_link

* Connector: Relational Database

Database Type: Data warehouse

* Database Server: dws-demo.dws.myhuaweiclouds.(Select

* Port: 8000

* Database Name: demo

* Username: dbadmin

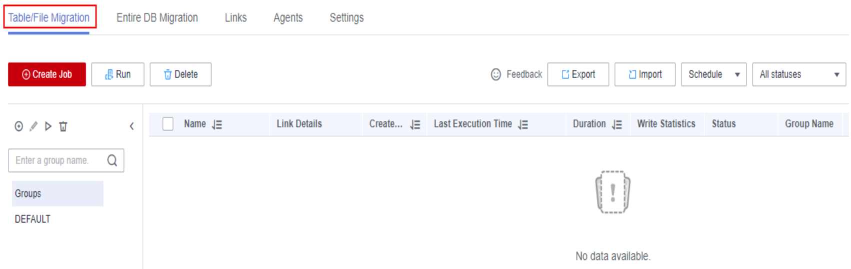
* Password:

Use Agent: Yes [✓ No]

Show Advanced Attributes

Step 6 After the links are created, click the **Table/File Migration** tab and then **Create Job**.

Figure 3-15 Creating a job



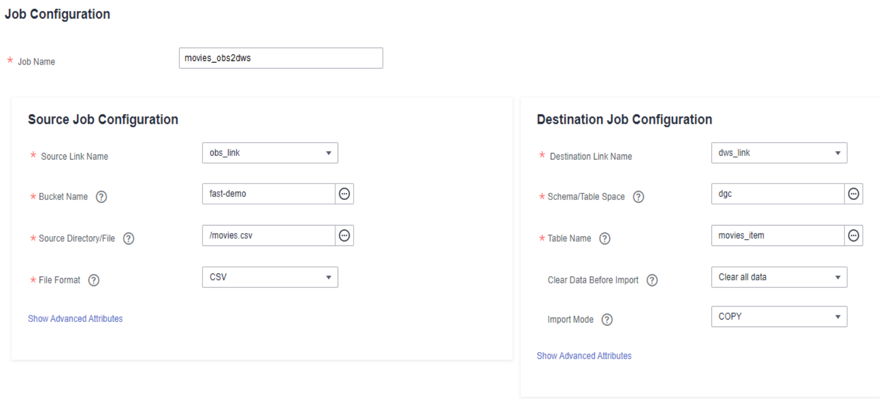
Step 7 Perform the following steps to configure job parameters:

1. As shown in **Figure 3-16**, enter **movies_obs2dws** for **Job Name** and configure the source job and destination job parameters.

NOTE

In this example, **Yes** is selected for **Clear Data Before Import**, indicating that data is cleared before data import each time the job is executed. Exercise caution when setting this parameter to avoid data loss.

Figure 3-16 Configuring job parameters



Job Configuration

* Job Name: movies_obs2dws

Source Job Configuration

- * Source Link Name: obs_link
- * Bucket Name: fast-demo
- * Source Directory/File: /movies.csv
- * File Format: CSV

Show Advanced Attributes

Destination Job Configuration

- * Destination Link Name: dws_link
- * Schema/Table Space: dgc
- * Table Name: movies_item
- Clear Data Before Import: Clear all data
- Import Mode: COPY

Show Advanced Attributes

2. In the **Source Job Configuration** and **Destination Job Configuration** areas, click **Show Advanced Attributes**. In the **Advanced Attributes** area, default values are provided. Set the parameters based on the actual data format.

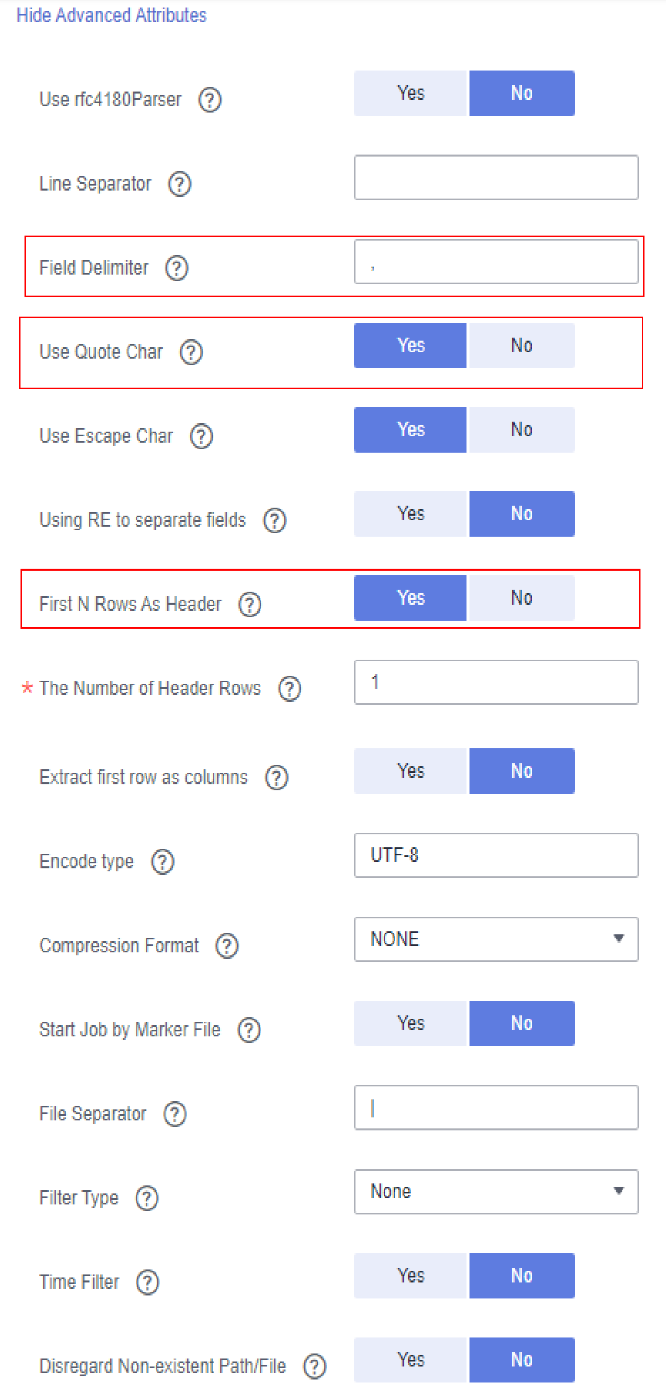
   In this example, pay attention to the following parameters in the advanced attributes of the source job and retain the default values for other parameters. You do not need to configure the advanced attributes of the destination job.

   – **Field Delimiter**: Retain the default value, which is a comma (,).

   – **Use Quote Char**: Select **Yes** because some original IMDbURL data contains commas (,).

   – **First N Rows As Header**: The default value is **No**. In this section, set this parameter to **Yes** and set **The Number of Header Rows** to **1**.

Figure 3-17 Source job advanced attributes



Hide Advanced Attributes

Use rfc4180Parser ? Yes [✓ No]

Line Separator ?

Field Delimiter ? ,

Use Quote Char ? [✓ Yes] No

Use Escape Char ? [✓ Yes] No

Using RE to separate fields ? Yes [✓ No]

First N Rows As Header ? [✓ Yes] No

* The Number of Header Rows ? 1

Extract first row as columns ? Yes [✓ No]

Encode type ? UTF-8

Compression Format ? NONE ▾

Start Job by Marker File ? Yes [✓ No]

File Separator ? |

Filter Type ? None ▾

Time Filter ? Yes [✓ No]

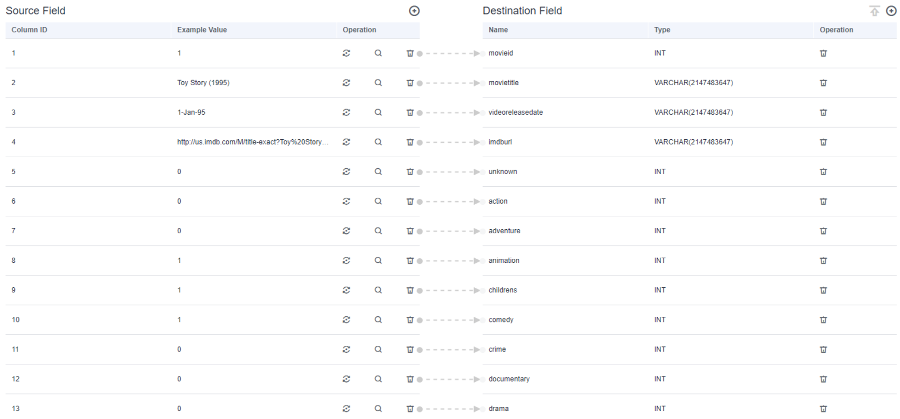
Disregard Non-existent Path/File ? Yes [✓ No]

3. Click **Next**, configure field mapping, and click **Next**.

   Map Field: In this example, you do not need to adjust the field mapping because the sequence of the source fields is the same as that of the destination fields.

   If they are different, you need to map the source fields with the destination fields by meaning. To map a field with another, move the cursor to the arrow start point of the source field. When **+** is displayed, hold down the mouse left button, move the cursor to the field with the same meaning as the source field, and release the mouse button.

Figure 3-18 Field mapping



Source Field

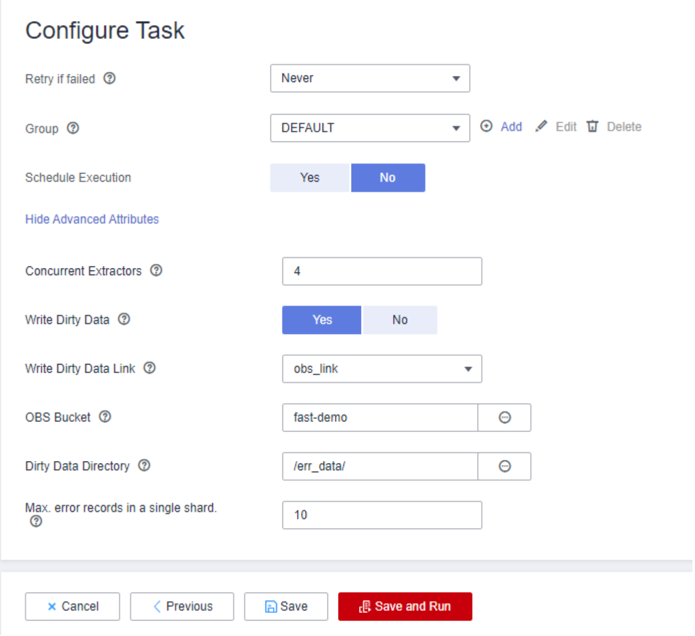
Column ID	Example Value	Operation
1	1	
2	Toy Story (1995)	
3	1-Jan-95	
4	http://us.imdb.com/M/title-exact?Toy%20Story...	
5	0	
6	0	
7	0	
8	1	
9	1	
10	1	
11	0	
12	0	
13	0	

Destination Field

Name	Type	Operation
movieid	INT	
movietitle	VARCHAR(2147483647)	
videoreleasedate	VARCHAR(2147483647)	
imdburl	VARCHAR(2147483647)	
unknown	INT	
action	INT	
adventure	INT	
animation	INT	
childrens	INT	
comedy	INT	
crime	INT	
documentary	INT	
drama	INT	

4. Configure **Retry upon Failure** , **Schedule Execution**, and advanced attributes. In this example, set **Write Dirty Data** to **Yes** and retain the default values for other parameters.

Figure 3-19 Configuring the task



Configure Task

Retry if failed	Never
Group	DEFAULT
Schedule Execution	Yes [✓ No]
Concurrent Extractors	4
Write Dirty Data	[✓ Yes] No
Write Dirty Data Link	obs_link
OBS Bucket	fast-demo
Dirty Data Directory	/err_data/
Max. error records in a single shard.	10

Click **Show Advanced Attributes** and set **Concurrent Extractors** and **Write Dirty Data**.

- **Concurrent Extractors**: Enter the number of extractors to be concurrently executed. The value range is 1 to 1000. If the value is too large, the extractors are queued.

  The number of concurrent extractors in a CDM migration job is related to the cluster specifications and table size.

  - You are advised to set this parameter to **4** for each CU (1 CPU and 4 GB) based on the cluster specifications.

- If each row of the table contains less than or equal to 1 MB data, you can extract data concurrently. If each row contains more than 1 MB data, you are advised to extract data in a single thread.

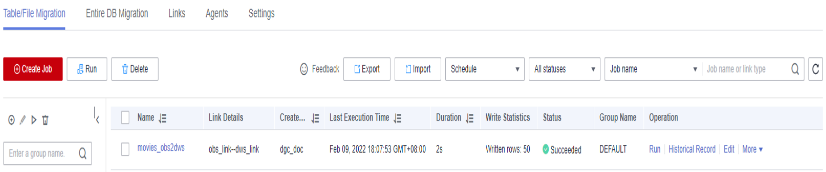
– **Write Dirty Data**: You are advised to set this parameter to **Yes** and set related parameters by referring to **Figure 3-19**. Dirty data refers to the data that does not match the destination fields and can be saved in a specified OBS bucket. If you select **Yes**, normal data can be written to the destination end, and the migration job will not be interrupted by the dirty data.

In this example, set **OBS Bucket** to **fast-demo** created in **Preparing Data Sources**. Go to the OBS console, create a directory, for example, **err_data**, in the bucket, and set **Dirty Data Directory** to this directory.

Step 8 Click **Save and Run**.

On the **Table/File Migration** page, you can view the created job.

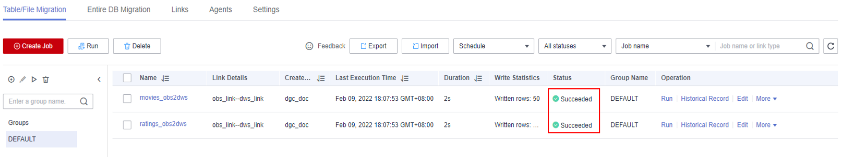
Figure 3-20 Execution result of the migration job



Name	Link Details	Create...	Last Execution Time	Duration	Write Statistics	Status	Group Name	Operation
movies_obs2dws	obs_link--dws_link	dgc_doc	Feb 09, 2022 18:07:53 GMT+08:00	2s	Written rows: 50	Succeeded	DEFAULT	Run | Historical Record | Edit | More ▾

Step 9 Repeat steps **Step 6** to **Step 8** to create another migration job named **ratings_obs2dws** for migrating data in the **ratings.csv** file to the **ratings_item** table of DWS. After the job is successfully executed, the data migration is complete.

Figure 3-21 Data migration result



Name	Link Details	Create...	Last Execution Time	Duration	Write Statistics	Status	Group Name	Operation
movies_obs2dws	obs_link--dws_link	dgc_doc	Feb 09, 2022 18:07:53 GMT+08:00	2s	Written rows: 50	Succeeded	DEFAULT	Run | Historical Record | Edit | More ▾
ratings_obs2dws	obs_link--dws_link	dgc_doc	Feb 09, 2022 18:07:53 GMT+08:00	2s	Written rows: ...	Succeeded	DEFAULT	Run | Historical Record | Edit | More ▾

Step 10 After the data migration is complete, you can go to the **DataArts Factory** page, create a DWS SQL script, and run the following SQL statements to check whether the data in the **movies_item** and **ratings_item** tables meets expectations:

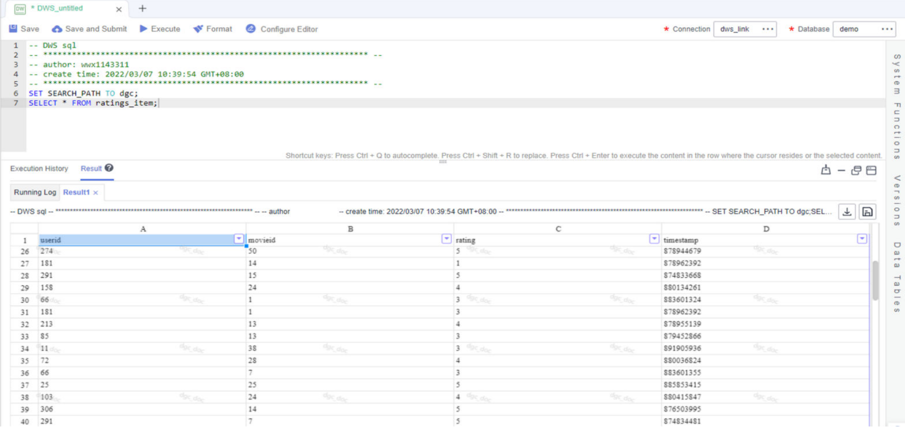
- Check the data in the **movies_item** table.

```
SET SEARCH_PATH TO dgc;
SELECT * FROM movies_item;
```

- Check the data in the **ratings_item** table.

```
SET SEARCH_PATH TO dgc;
SELECT * FROM ratings_item;
```

Figure 3-22 Viewing data in DWS tables



```
-- DWS sql
-- ********************************************************************** --
-- author: wwx1143311
-- create time: 2022/03/07 10:39:54 GMT+08:00
-- ********************************************************************** --
SET SEARCH_PATH TO dgc;
SELECT * FROM ratings_item;
```

Shortcut keys: Press Ctrl + Q to autocomplete. Press Ctrl + Shift + R to replace. Press Ctrl + Enter to execute the content in the row where the cursor resides or the selected content.

-- DWS sql -- ** -- -- author -- create time: 2022/03/07 10:39:54 GMT+08:00 -- ** -- SET SEARCH_PATH TO dgc;SEL...

	A	B	C	D
1	userid	movieid	rating	timestamp
26	274	50	5	878944679
27	181	14	1	878962392
28	291	15	5	874833668
29	158	24	4	880134261
30	66	1	3	883601324
31	181	1	3	878962392
32	213	13	4	878955139
33	85	13	3	879452866
34	11	38	3	891905936
35	72	28	4	880036824
36	66	7	3	883601355
37	25	25	5	885853415
38	103	24	4	880415847
39	306	14	5	876503995
40	291	7	5	874834481

----End

3.4 Step 3: Develop Data

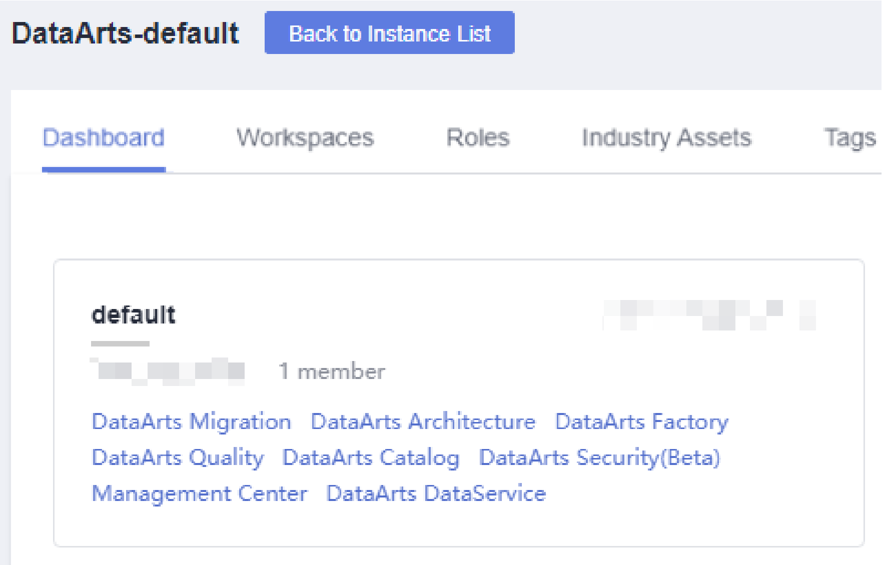
This step describes how to use the movie information and rating data to analyze 10 top-rated movies and 10 most frequently scored movies. Jobs are periodically executed and the results are exported to tables every day for data analysis.

Creating DWS SQL Script top_rating_movie for Storing 10 Top-rated Movies

The method of finding out the 10 top-rated movies is as follows: Calculate the total score of each movie and the number of the users who participate in scoring the movies, filter out the movies that are scored by less than three users, and then return the movie names, average scores, and participant quantity.

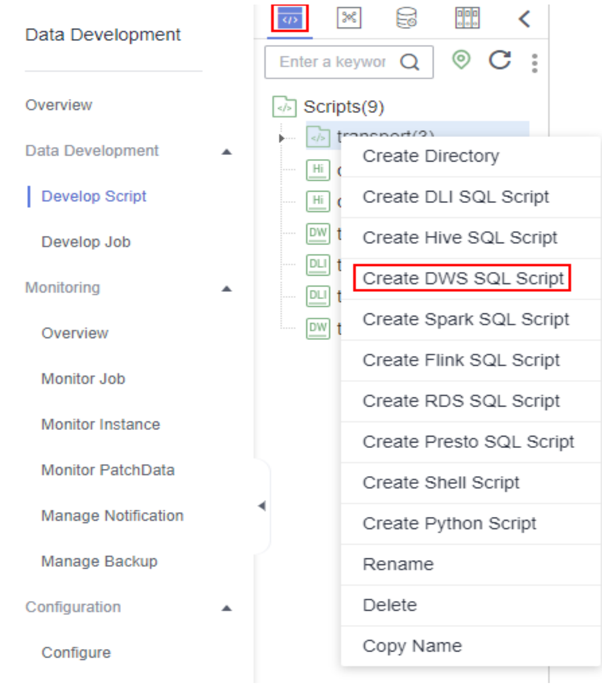
Step 1 Log in to the DataArts Studio console. Locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Factory**.

Figure 3-23 DataArts Factory



Step 2 Create a DWS SQL script used to create data tables by entering DWS SQL statements in the editor.

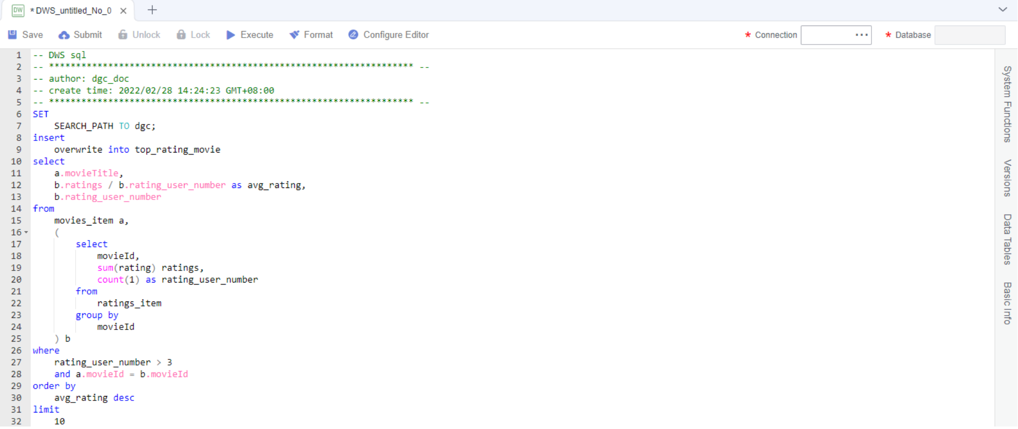
Figure 3-24 Creating a script



Step 3 In the SQL editor, enter the following SQL statements and click **Execute** to calculate the 10 top-rated movies from the **movies_item** and **ratings_item** tables and save the result to the **top_rating_movie** table.

```
SET
    SEARCH_PATH TO dgc;
insert
    overwrite into top_rating_movie
select
    a.movieTitle,
    b.ratings / b.rating_user_number as avg_rating,
    b.rating_user_number
from
    movies_item a,
    (
        select
            movieId,
            sum(rating) ratings,
            count(1) as rating_user_number
        from
            ratings_item
        group by
            movieId
    ) b
where
    rating_user_number > 3
    and a.movieId = b.movieId
order by
    avg_rating desc
limit
    10
```

Figure 3-25 Script (top_rating_movie)



```
-- DWS sql
-- ******************************************************************** --
-- author: dgc_doc
-- create time: 2022/02/28 14:24:23 GMT+08:00
-- ******************************************************************** --
SET
    SEARCH_PATH TO dgc;
insert
    overwrite into top_rating_movie
select
    a.movieTitle,
    b.ratings / b.rating_user_number as avg_rating,
    b.rating_user_number
from
    movies_item a,
    (
        select
            movieId,
            sum(rating) ratings,
            count(1) as rating_user_number
        from
            ratings_item
        group by
            movieId
    ) b
where
    rating_user_number > 3
    and a.movieId = b.movieId
order by
    avg_rating desc
limit
    10
```

The key parameters are as follows:

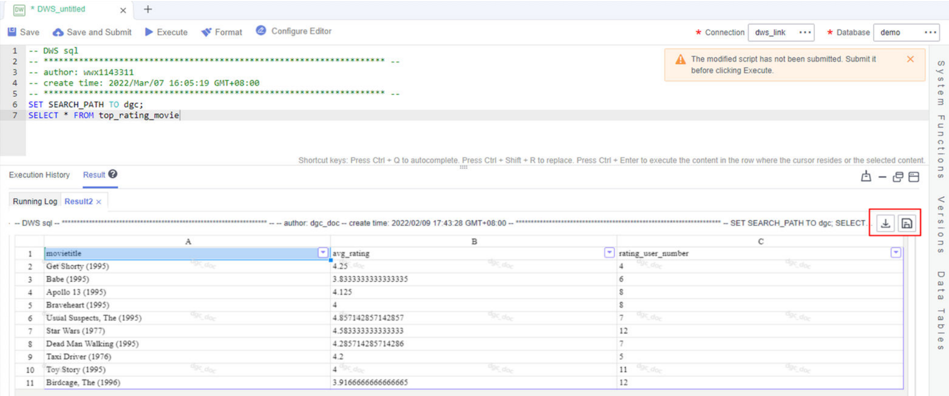
- **Data Connection**: DWS data connection created in **Step 3**
- **Database**: database created in **step 5**

Step 4 After debugging the script, click **Save and Submit** to submit the script and name it **top_rating_movie**. This script will be referenced later in **Developing and Scheduling a Job**.

Step 5 After the script is saved and executed successfully, you can use the following SQL statement to view data in the **top_rating_movie** table. You can also download or dump the table data by referring to **Figure 3-26**.

```
SET SEARCH_PATH TO dgc;
SELECT * FROM top_rating_movie
```

Figure 3-26 Viewing the data in the top_rating_movie table



```
-- DWS sql
-- ******************************************************************** --
-- author: wwx1143311
-- create time: 2022/Mar/07 16:05:19 GMT+08:00
-- ******************************************************************** --
SET SEARCH_PATH TO dgc;
SELECT * FROM top_rating_movie
```

The modified script has not been submitted. Submit it before clicking Execute.

Shortcut keys: Press Ctrl + Q to autocomplete. Press Ctrl + Shift + R to replace. Press Ctrl + Enter to execute the content in the row where the cursor resides or the selected content.

-- DWS sql -- ** -- -- author: dgc_doc -- create time: 2022/02/09 17:43:28 GMT+08:00 -- ** -- SET SEARCH_PATH TO dgc; SELECT

	A	B	C
1	movietitle	avg_rating	rating_user_number
2	Get Shorty (1995)	4.25	4
3	Babe (1995)	3.8333333333333335	6
4	Apollo 13 (1995)	4.125	8
5	Braveheart (1995)	4	8
6	Usual Suspects, The (1995)	4.857142857142857	7
7	Star Wars (1977)	4.583333333333333	12
8	Dead Man Walking (1995)	4.285714285714286	7
9	Taxi Driver (1976)	4.2	5
10	Toy Story (1995)	4	11
11	Birdcage, The (1996)	3.9166666666666665	12

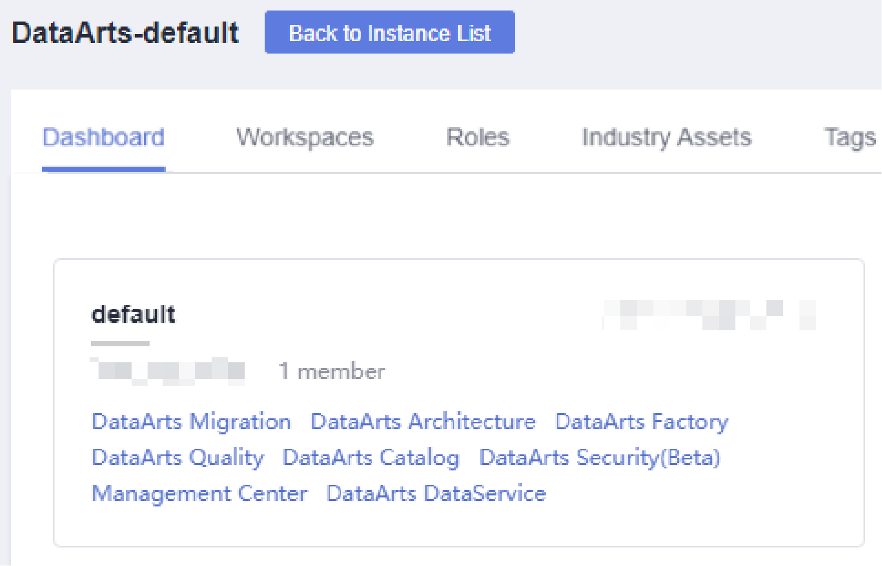
----End

Creating DWS SQL Script top_active_movie for Storing 10 Most Frequently Scored Movies

The method of finding out the 10 most frequently scored movies is as follows: Calculate the 10 most frequently scored movies whose average scores are higher than 3.5.

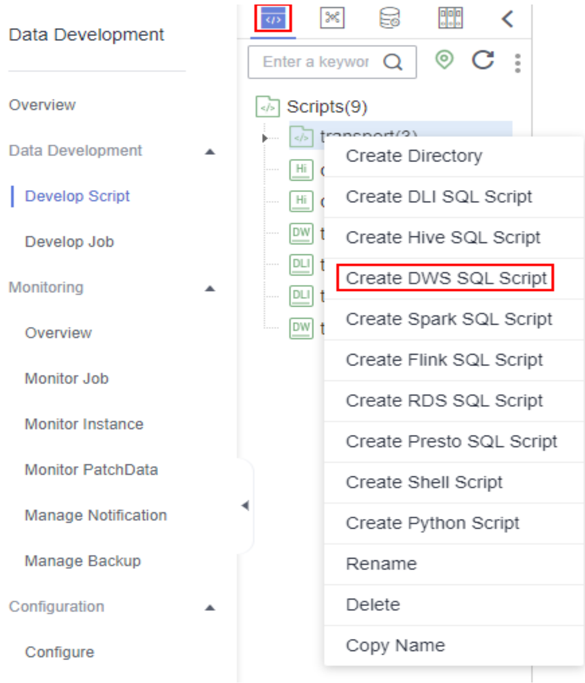
Step 1 Log in to the DataArts Studio console. Locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Factory**.

Figure 3-27 DataArts Factory



Step 2 Create a DWS SQL script used to create data tables by entering DWS SQL statements in the editor.

Figure 3-28 Creating a script

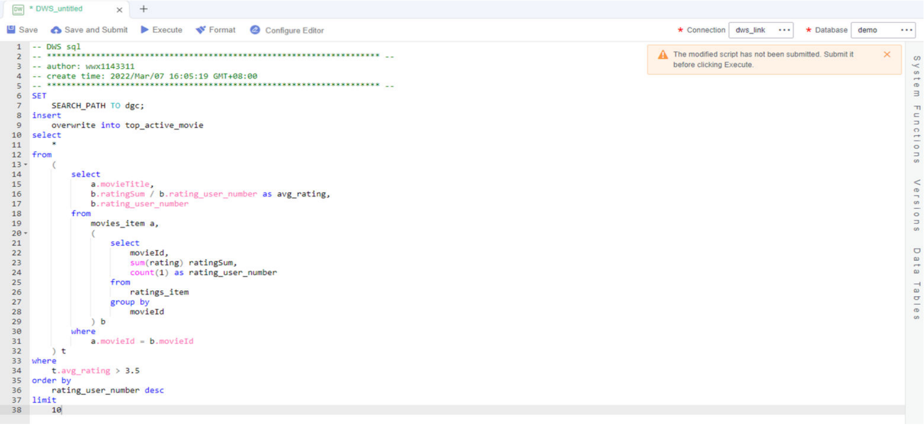


Step 3 In the SQL editor, enter the following SQL statements and click **Execute** to calculate the 10 most frequently scored movies from the **movies_item** and **ratings_item** tables and save the result to the **top_active_movie** table.

```
SET
   SEARCH_PATH TO dgc;
```

```
insert
    overwrite into top_active_movie
select
    *
from
    (
        select
            a.movieTitle,
            b.ratingSum / b.rating_user_number as avg_rating,
            b.rating_user_number
        from
            movies_item a,
            (
                select
                    movieId,
                    sum(rating) ratingSum,
                    count(1) as rating_user_number
                from
                    ratings_item
                group by
                    movieId
            ) b
        where
            a.movieId = b.movieId
    ) t
where
    t.avg_rating > 3.5
order by
    rating_user_number desc
limit
    10
```

Figure 3-29 Script (top_active_movie)



* Connection: dws_link

* Database: demo

The modified script has not been submitted. Submit it before clicking Execute.

```
-- DWS sql
-- ******************************************************************** --
-- author: wwx1143311
-- create time: 2022/Mar/07 16:05:19 GMT+08:00
-- ******************************************************************** --
SET
    SEARCH_PATH TO dgc;
insert
    overwrite into top_active_movie
select
    *
from
    (
        select
            a.movieTitle,
            b.ratingSum / b.rating_user_number as avg_rating,
            b.rating_user_number
        from
            movies_item a,
            (
                select
                    movieId,
                    sum(rating) ratingSum,
                    count(1) as rating_user_number
                from
                    ratings_item
                group by
                    movieId
            ) b
        where
            a.movieId = b.movieId
    ) t
where
    t.avg_rating > 3.5
order by
    rating_user_number desc
limit
    10
```

The key parameters are as follows:

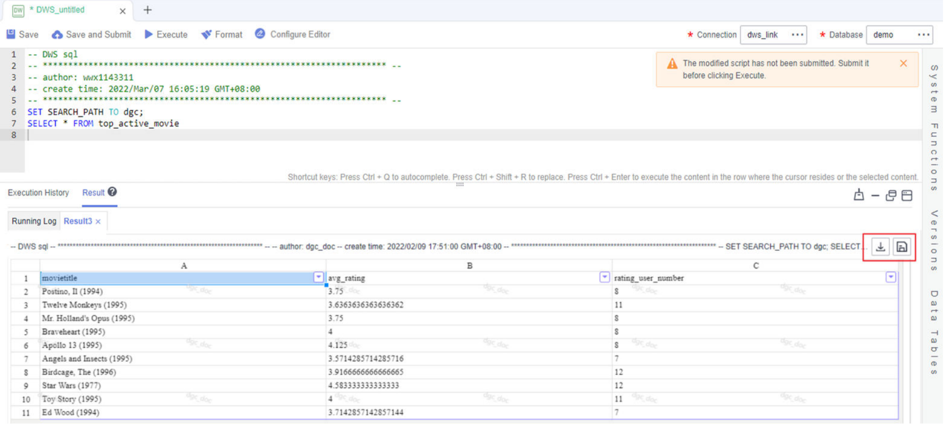
- **Data Connection**: DWS data connection created in **Step 3**
- **Database**: database created in **step 5**

Step 4 After debugging the script, click **Save and Submit** to submit the script and name it **top_active_movie**. This script will be referenced later in **Developing and Scheduling a Job**.

Step 5 After the script is saved and executed successfully, you can use the following SQL statement to view data in the **top_active_movie** table. You can also download or dump the table data by referring to **Figure 3-30**.

```
SET SEARCH_PATH TO dgc;
SELECT * FROM top_active_movie
```

Figure 3-30 Viewing the data in the top_active_movie table



The modified script has not been submitted. Submit it before clicking Execute.

```
-- DWS sql
-- ********************************************************************** --
-- author: wwx1143311
-- create time: 2022/Mar/07 16:05:19 GMT+08:00
-- ********************************************************************** --
SET SEARCH_PATH TO dgc;
SELECT * FROM top_active_movie
```

Shortcut keys: Press Ctrl + Q to autocomplete. Press Ctrl + Shift + R to replace. Press Ctrl + Enter to execute the content in the row where the cursor resides or the selected content.

-- DWS sql -- ** -- -- author: dgc_doc -- create time: 2022/02/09 17:51:00 GMT+08:00 -- ** -- SET SEARCH_PATH TO dgc; SELECT...

	A	B	C
1	movietitle	avg_rating	rating_user_number
2	Postino, Il (1994)	3.75	8
3	Twelve Monkeys (1995)	3.6363636363636362	11
4	Mr. Holland's Opus (1995)	3.75	8
5	Braveheart (1995)	4	8
6	Apollo 13 (1995)	4.125	8
7	Angels and Insects (1995)	3.5714285714285716	7
8	Birdcage, The (1996)	3.9166666666666665	12
9	Star Wars (1977)	4.583333333333333	12
10	Toy Story (1995)	4	11
11	Ed Wood (1994)	3.7142857142857144	7

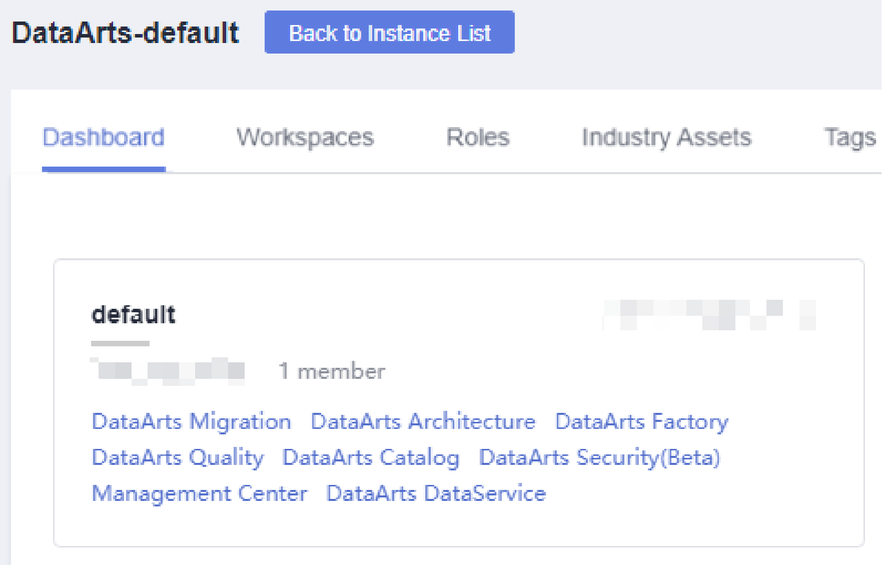
----End

Developing and Scheduling a Job

Assume that the **movie** and **rating** tables in the OBS bucket are changing in real time. To update top 10 movies every day, use the job orchestration and scheduling functions of DataArts Factory.

Step 1 Log in to the DataArts Studio console. Locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Factory**.

Figure 3-31 DataArts Factory



default [redacted]

[redacted] 1 member

Step 2 Create a batch job named **topmovie**.

Figure 3-32 Creating a job

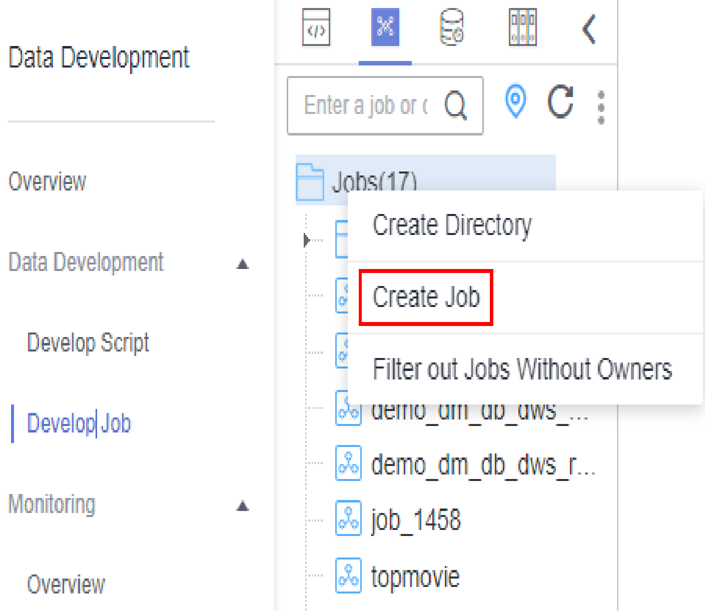
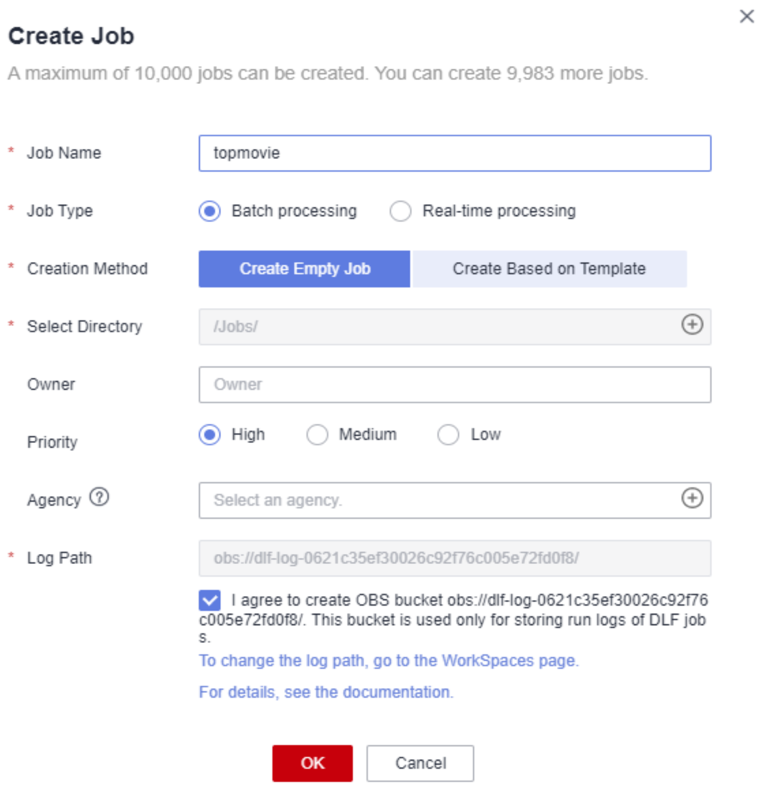


Figure 3-33 Configuring the job



Create Job

A maximum of 10,000 jobs can be created. You can create 9,983 more jobs.

* Job Name	topmovie
* Job Type	● Batch processing ○ Real-time processing
* Creation Method	Create Empty Job / Create Based on Template
* Select Directory	/Jobs/
Owner	Owner
Priority	● High ○ Medium ○ Low
Agency	Select an agency.
* Log Path	obs://dlf-log-0621c35ef30026c92f76c005e72fd0f8/

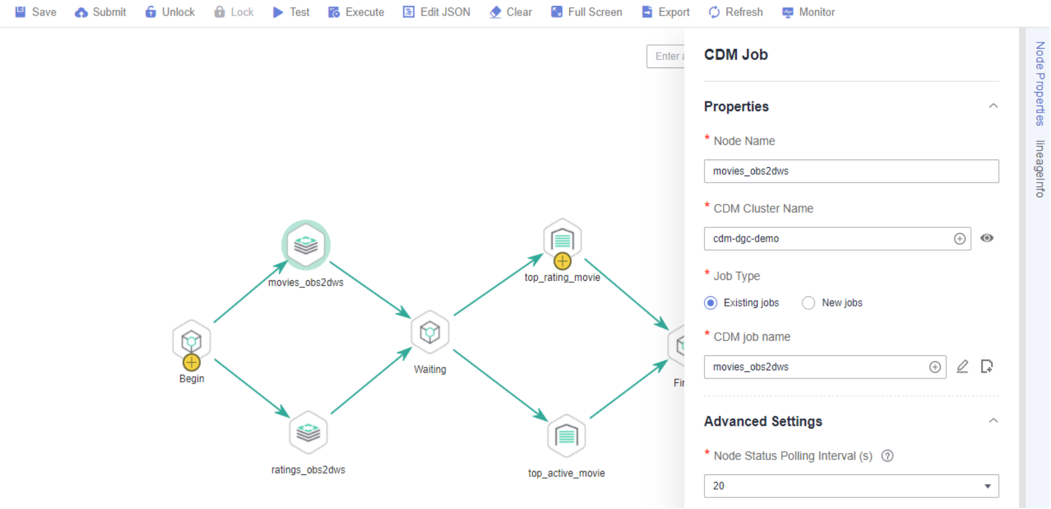
☑ I agree to create OBS bucket obs://dlf-log-0621c35ef30026c92f76c005e72fd0f8/. This bucket is used only for storing run logs of DLF jobs.

To change the log path, go to the WorkSpaces page.

For details, see the documentation.

Step 3 Open the created job, drag two CDM Job nodes, three Dummy nodes, and two DWS SQL nodes to the canvas, select and drag ⊕, and orchestrate the job shown in **Figure 3-34**.

Figure 3-34 Connecting nodes and configuring node properties



CDM Job

Properties

* Node Name: movies_obs2dws

* CDM Cluster Name: cdm-dgc-demo

* Job Type: ● Existing jobs ○ New jobs

* CDM job name: movies_obs2dws

Advanced Settings

* Node Status Polling Interval (s): 20

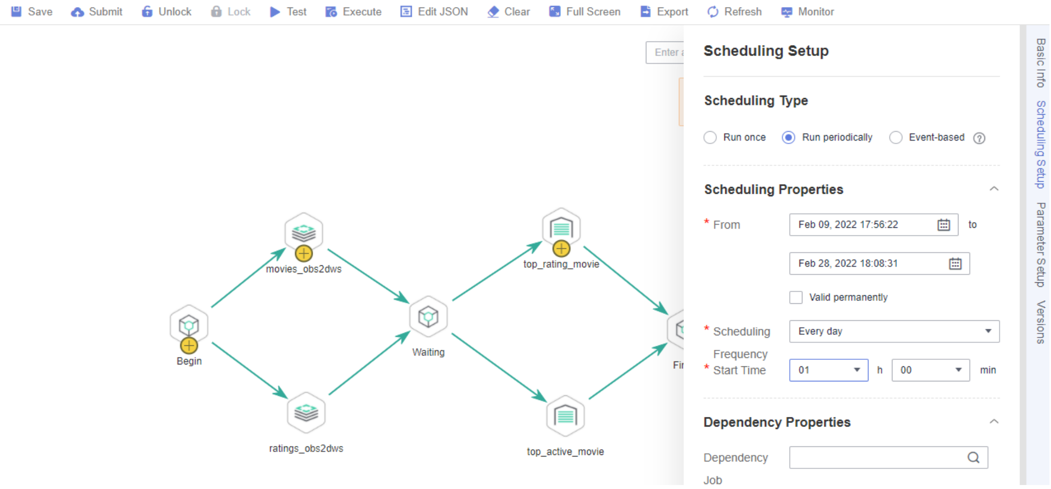
Key nodes:

- **Begin** (Dummy node): serves only as a start identifier.
- **movies_obs2dws** (CDM Job node): In **Node Properties**, select the CDM cluster in **Step 2: Integrate Data** and associate it with the CDM job **movies_obs2dws**.
- **ratings_obs2dws** (CDM Job node): In **Node Properties**, select the CDM cluster in **Step 2: Integrate Data** and associate it with the CDM job **ratings_obs2dws**.
- **Waiting** (Dummy node): No operation is performed. It is an identifier of the execution completion of the previous node.
- **top_rating_movie** (DWS SQL node): In **Node Properties**, associate this node with the DWS SQL script **top_rating_movie** you have created in **Creating DWS SQL Script top_rating_movie**.
- **top_active_movie** (DWS SQL node): In **Node Properties**, associate this node with the DWS SQL script **top_active_movie** you have created in **Creating DWS SQL Script top_active_movie**.
- **Finish** (Dummy node): serves only as an end identifier.

Step 4 After configuring the job, click ▶ to test it.

Step 5 If the job runs properly, click **Scheduling Setup** in the right pane and configure the scheduling policy for the job.

Figure 3-35 Configuring scheduling



Scheduling Setup

Scheduling Type

○ Run once ● Run periodically ○ Event-based

Scheduling Properties

* From: Feb 09, 2022 17:56:22 to Feb 28, 2022 18:08:31

□ Valid permanently

* Scheduling Frequency: Every day

* Start Time: 01 h 00 min

Dependency Properties

Dependency Job:

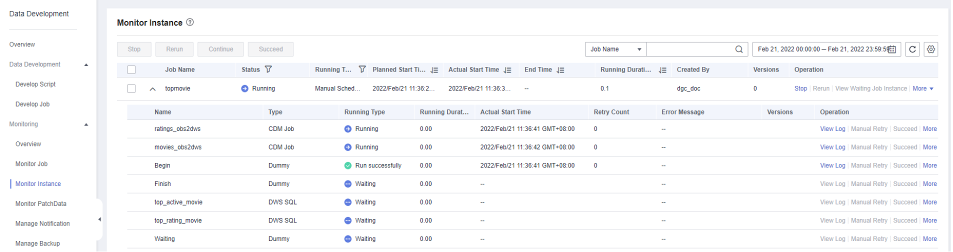
Notes:

- **Scheduling Properties**: The job is executed at 01:00 every day from Feb 09 to Feb 28, 2022.
- **Dependency Properties**: You can configure a dependency job for this job. You do not need to configure it in this practice.
- **Cross-Cycle Dependency**: Select **Independent on the previous schedule cycle**.

Step 6 Click **Save and Submit** and **Execute**. Then the job will be automatically executed every day so the 10 highest scored and most frequently scored movies are automatically saved to the **top_active_movie** and **top_rating_movie** tables, respectively.

Step 7 If you want to check the job execution result, choose **Monitoring** > **Monitor Instance** in the left navigation pane.

Figure 3-36 Viewing the job execution status



Monitor Instance

Job Name: | Feb 21, 2022 00:00:00 – Feb 21, 2022 23:59:59

Job Name	Status	Running T...	Planned Start Ti...	Actual Start Time	End Time	Running Durati...	Created By	Versions	Operation
topmovie	Running	Manual Sched...	2022/Feb/21 11:36:2...	2022/Feb/21 11:36:3...	--	0.1	dgc_doc	0	Stop | Rerun | View Waiting Job Instance | More

Name	Type	Running Type	Running Durat...	Actual Start Time	Retry Count	Error Message	Versions	Operation
ratings_obs2dws	CDM Job	Running	0.00	2022/Feb/21 11:36:41 GMT+08:00	0	--		View Log | Manual Retry | Succeed | More
movies_obs2dws	CDM Job	Running	0.00	2022/Feb/21 11:36:42 GMT+08:00	0	--		View Log | Manual Retry | Succeed | More
Begin	Dummy	Run successfully	0.00	2022/Feb/21 11:36:41 GMT+08:00	0	--		View Log | Manual Retry | Succeed | More
Finish	Dummy	Waiting	0.00	--		--		View Log | Manual Retry | Succeed | More
top_active_movie	DWS SQL	Waiting	0.00	--		--		View Log | Manual Retry | Succeed | More
top_rating_movie	DWS SQL	Waiting	0.00	--		--		View Log | Manual Retry | Succeed | More
Waiting	Dummy	Waiting	0.00	--		--		View Log | Manual Retry | Succeed | More

----End

You can also configure notifications to be sent through SMS messages, emails, or console when a job encounters exceptions or fails.

Now you have learned the data integration and development process based on movie scores. In addition, you can analyze the ratings and browsing of different types of movies to provide valuable information for marketing decision-making, advertising, and user behavior prediction.

3.5 (Optional) Step 4: Unsubscribe from Services

In this development scenario, DataArts Studio, OBS, and GaussDB(DWS) incur fees. If you configure notifications, you may be billed for the following service:

- SMN: If you enable SMN notifications for your DataArts Studio modules, you need to pay for the notifications.
- EIP: If you buy an EIP for your CDM cluster, you need to pay for the EIP.
- DEW: If you enable KMS when creating a link in DataArts Migration or creating a connection in Management Center, you will be billed for key management.

After the development is complete, unsubscribe from DataArts Studio and other related services and delete resources in a timely manner to avoid undesired fees.

Table 3-5 Unsubscription methods for services

Service	Billing	Unsubscription Method
DataArts Studio	**DataArts Studio Billing**	DataArts Studio instances support only the yearly/monthly billing mode. You can unsubscribe from the package by referring to **Unsubscriptions**.
OBS	**OBS Billing**	OBS supports pay-per-use and yearly/monthly billing modes. Packages cannot be unsubscribed. In this example, the pay-per-use billing mode is used. You can delete the created bucket after using it. In addition, DataArts Studio job logs and DLI dirty data are stored in an OBS bucket named **dlf-log-{***Project id***}** by default. You can delete the bucket after unsubscribing from DataArts Studio.
GaussDB(DWS)	**GaussDB(DWS) Billing**	GaussDB(DWS) supports pay-per-use and yearly/monthly billing modes. In this example, the pay-per-use billing mode is used. You can delete the GaussDB(DWS) cluster after you finish with it. If you chose the yearly/monthly billing mode, you can unsubscribe from the yearly/monthly package you bought and delete the GaussDB(DWS) cluster after you finish with it by referring to **Unsubscriptions**.
SMN	**SMN Billing**	You pay only for what you use. After you unsubscribe from DataArts Studio, no notification will be generated. You can also delete the topics and subscriptions that have been generated.
EIP	**EIP Billing**	EIP supports the pay-per-use and yearly/monthly billing modes. In this example, the pay-per-use billing mode is used. You can release the EIP after you finish with it. If you chose the yearly/monthly billing mode, you can unsubscribe from the yearly/monthly package you bought and release the EIP after you finish with it by referring to **Unsubscriptions**.

Service	Billing	Unsubscription Method
DEW	**DEW Billing**	KMS keys are billed pay per use. You can delete the KMS keys generated by DEW.

4 Advanced Users: Data Governance Based on Taxi Trip Data

4.1 Example Scenario

This getting-started guide describes how to complete end-to-end data operations on DataArts Studio.

In this case, MRS Hive is used as the data lake foundation, and DataArts Studio is used for end-to-end governance of taxi trip data of a city. The following objectives are expected to be achieved through data governance:

- Standardized data and models
- Unified statistics standards and high-quality data reports
- Data quality monitoring and alarm
- Daily revenue statistics
- Monthly revenue statistics
- Statistics on the revenue proportion of each payment type

Process Overview

You can govern data in the example scenario based on the process in **Table 4-1**.

Table 4-1 Process of data governance using DataArts Studio

Process	Description	Subtask	Operation
Step 1: Process Design	Before using DataArts Studio, conduct a service survey and requirement analysis.	Requirement analysis, service survey, and process design	**Requirement Analysis** **Service Survey**

Process	Description	Subtask	Operation
Step 2: Preparations	If you are new to DataArts Studio, create a DataArts Studio instance and a workspace.	Preparations before using DataArts Studio	**Preparations**
Step 3: DataArts Migration	Use DataArts Studio to upload data from data sources to the cloud. You can migrate offline or historical data. DataArts Migration can migrate a single table, file, entire database, and incremental data. You can use it to migrate data between homogeneous and heterogeneous data sources such as on-premises and cloud-based file systems, relational databases, data warehouses, NoSQL databases, big data services, and object storage.	Data integration	**Creating a Cluster** **Creating Source and Destination Links for Data Migration** **Creating a Table/ File Migration Job**
Step 4: Metadata Collection	Collect metadata of raw data for data management and monitoring.	Metadata collection	**Collecting and Monitoring Metadata**
Step 5: 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.	Preparations	**Adding Reviewers** **Configuration Center Management**
		Subject design	**Designing a Subject**
		Standard management	**Creating and Publishing Lookup Tables** **Creating and Publishing Data Standards**
		ER modeling	**Creating Two ER Models for the SDI and DWI Layers**

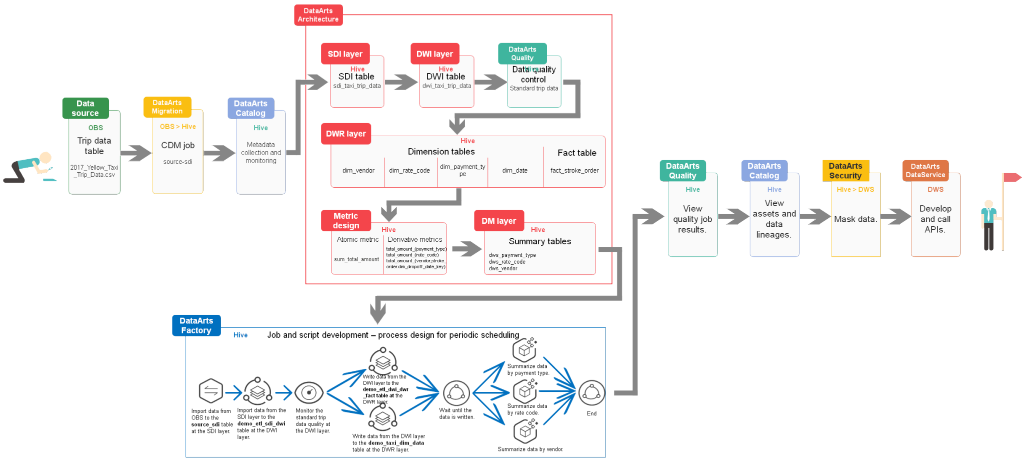
Process	Description	Subtask	Operation
		Dimensional modeling	Creating and Publishing Dimensions for the DWR Layer Creating and Publishing a Fact Table for the DWR Layer
		Metric design	Creating and Publishing Technical Metrics
		Data mart building	Creating and Publish Summary Tables for the DM Layer
Step 6: DataArts Factory	Use DataArts Factory to manage diverse big data services. DataArts Studio enables a variety of operations such as data management, script development, job development, job scheduling, O&M, and monitoring, facilitating data analysis and processing.	Data management	Managing data
		Script development	Developing a Script
		Job development .	Developing a Batch Job
		O&M scheduling	O&M Scheduling
Step 7: DataArts Quality	Use DataArts Quality to monitor metrics. 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 Studio uses automatically generated quality rules to standardize data, and supports periodic monitoring.	Metric monitoring	Monitoring Business Metrics
		Data quality control	Viewing Quality Jobs

Process	Description	Subtask	Operation
Step 8: DataArts Catalog	In the DataArts Catalog module, you can view data maps.	Data maps	**Viewing Logical Assets and Technical Assets**
(Optional) Step 9: Service Unsubscription	Unsubscribe from the service to avoid unnecessary billing.	Unsubscribing from the service	**(Optional) Unsubscribing from the Service**

4.2 Step 1: Process Design

This guide uses the collection of operations statistics from a taxi vendor in 2017 as an example. **Figure 4-1** shows the data governance process which is based on **requirement analysis** and **service survey**.

Figure 4-1 Process design


Data source
OBS
Trip data table
2017_Yellow_Taxi_Trip_Data.csv
DataArts Migration
OBS > Hive
CDM job
source-sdi
DataArts Catalog
Hive
Metadata collection and monitoring
DataArts Architecture
SDI layer
Hive
SDI table
sdi_taxi_trip_data
DWI layer
Hive
DWI table
dwi_taxi_trip_data
DataArts Quality
Hive
Data quality control
Standard trip data
DWR layer
Hive
Dimension tables
dim_vendor
dim_rate_code
dim_payment_type
dim_date
Fact table
fact_stroke_order
Metric design
Hive
Atomic metric
sum_total_amount
Derivative metrics
total_amount_(payment_type)
total_amount_(rate_code)
total_amount_(vendor,stroke_order.dim_dropoff_date_key)
DM layer
Hive
Summary tables
dws_payment_type
dws_rate_code
dws_vendor
DataArts Factory
Hive
Job and script development – process design for periodic scheduling
Import data from OBS to the source_sdi table at the SDI layer.
Import data from the SDI layer to the demo_etl_sdi_dwi table at the DWI layer.
Monitor the standard trip data quality at the DWI layer.
Write data from the DWI layer to the demo_etl_dwi_dwr_fact table at the DWR layer.
Write data from the DWI layer to the demo_taxi_dim_data table at the DWR layer.
Wait until the data is written.
Summarize data by payment type.
Summarize data by rate code.
Summarize data by vendor.
End
DataArts Quality
Hive
View quality job results.
DataArts Catalog
Hive
View assets and data lineages.
DataArts Security
Hive > DWS
Mask data.
DataArts DataService
DWS
Develop and call APIs.

Requirement Analysis

Requirement analysis helps you develop a data governance framework to support the process design for data governance.

In this example scenario, the following data problems exist:

- No standardized model is available.
- There is no standard for data field naming.
- Data content is not standard, and data quality is uncontrollable.
- Statistics standards are inconsistent, hindering business decision-making.

With data governance of DataArts Studio, we expect to achieve the following objectives:

- Standardized data and models
- Unified statistics standards and high-quality data reports
- Data quality monitoring and alarm

- Daily revenue statistics
- Monthly revenue statistics
- Statistics on the revenue proportion of each payment type

Service Survey

Before using DataArts Studio, conduct a service survey to understand the component functions required in the service process and analyze the subsequent service load.

Table 4-2 Service survey table

No.	Configuration Item	Information to Be Collected	Survey Result	Remarks
1	Workspace	Organizations and relationships between the enterprise's big data departments	N/A	Properly plan workspaces to reduce the complexity of workspace dependency
		Access control permissions on data and resources between departments	N/A	User permissions and resource permissions control are involved.
2	DataArts Migration	Data source from which the data is to be migrated and the data source version	CSV source data files in the OBS bucket	N/A
		Full data volume of each data source	2,114 bytes	N/A
		Daily incremental data volume of each data source	N/A	N/A
		Types and versions of data sources at the destination	MRS Hive 3.1	N/A
		Data migration period: day, hour, minute, or real-time	Day	N/A
		Network bandwidth between data sources at the source and destination	100 MB	N/A

No.	Configuration Item	Information to Be Collected	Survey Result	Remarks
		Description of the network connectivity between the data sources and integration tools	N/A	N/A
		Database migration: number of survey tables and maximum table size	N/A. In this example, data needs to be migrated from OBS to the database.	Understand the scale of database migration and whether the migration duration of the largest table is acceptable.
		File migration: number of files, and whether the size of any file reaches 1 TB	A CSV file smaller than 1 TB	N/A
3	DataArts Factory	Whether job orchestration and scheduling are required	Yes	N/A
		Services required in orchestration and scheduling, such as MRS, GaussDB(DWS), and CDM	DataArts Migration and DataArts Quality of DataArts Studio, and MRS Hive	Understand application scenarios of jobs to further investigate the suitability of platform capabilities for customer scenarios.
		Number of jobs	Less than 20	Understand the job scale. Generally, the job scale is described by the number of operators and can be estimated based on the number of tables.

No.	Configuration Item	Information to Be Collected	Survey Result	Remarks
		Number of times a job is scheduled	Unlimited	Determine the DataArts Studio edition based on the scheduling quota of each DataArts Studio sales edition.
		Number of data developers	1	N/A
4	DataArts Architecture	Data sources and number of tables	Only one CSV file	Analyze source data to understand the data source and overall situation.
		Services, requirements, and benefits	Standardize data and models and collect statistics on revenue in a flexible manner.	Analyze the destination to understand the purposes of data governance and digitalization.
		Data survey, data overview, data standards degree, and industry standards overview	N/A	Analyze the process to understand the standards and quality compliance in the data governance process.
5	DataArts Quality	Requirements and benefits	Data quality monitoring	Monitor more data sources and rules.

No.	Configuration Item	Information to Be Collected	Survey Result	Remarks
		Number of jobs	1	You can manually create dozens of jobs or enable the function of automatically generating data quality jobs on DataArts Architecture. If the API for creating data quality jobs is called, more than 100 quality jobs can be created.
		Application scenarios	Standardize and cleanse data at the DWI layer.	Generally, before and after data processing, the data quality is monitored from six dimensions. If any data that does not comply with rules is detected, users will receive an alarm notification.
6	DataArts Catalog	Data sources to support	MRS Hive	N/A
		Data volume	A table contains fewer than 100 records.	A maximum of 1 million tables can be managed.
		Scheduling frequency of metadata collection	N/A	Collection tasks can be executed by hour, day, or week.

No.	Configuration Item	Information to Be Collected	Survey Result	Remarks
		Key metrics of metadata collection	N/A	The key metrics include the table name, field name, owner, description, and creation time.
		Application scenarios of tags	N/A	Tags are highly related keywords that help you classify and describe assets to facilitate search.
7	DataArts Security	Data sources to which access is controlled	N/A	Access to the following components is controlled: HDFS, Hive, HBase, Yarn, Kafka, Storm, and Elasticsearch.
		Data security levels to be identified	N/A	A maximum of 10 data security levels can be defined.
		Data sources to be masked	In this example, data in the MRS standard trip table needs to be masked to DWS.	Only DWS and MRS data sources can be masked.
		Data sources that require watermarking	N/A	Watermarks can be embedded only for DWS and MRS data sources.

No.	Configuration Item	Information to Be Collected	Survey Result	Remarks
8	DataArts DataService	Open data sources	Revenue summary table	Generally, these data sources store the tables at the final layer after a data warehouse is established. Such tables contain high-quality service data but fewer records, which can be directly displayed.
		Daily data calls	N/A	If the database response takes a long period of time due to complex extraction logic, the data calling volume will decrease.
		Number of peak data calls per second	N/A	The number of peak data calls per second varies depending on the edition in use and data extraction logic.
		Average latency of a single data call	N/A	The database response duration is related to the data extraction logic.
		Whether data access records are required	N/A	N/A
		Data access method: intranet or Internet	N/A	N/A

No.	Configuration Item	Information to Be Collected	Survey Result	Remarks
		Number of DataArts DataService developers	1	N/A

4.3 Step 2: Preparations

Preparations Before Using DataArts Studio

If you use DataArts Studio for the first time, register a , buy a DataArts Studio instance, and create a workspace by following the instructions provided in **Preparations**. Then you can go to the created workspace and start using DataArts Studio.

In this example, the has all the permissions required for performing all the data operations on DataArts Studio so that the entire data governance process using DataArts Studio can be demonstrated.

Preparing a Data Source

This guide uses the collection of operations statistics from a taxi vendor in 2017 as an example. This example uses data from **https://data.cityofnewyork.us/Transportation/2017-Yellow-Taxi-Trip-Data/biws-g3hs**.

To facilitate demonstration, some sample data is provided to simulate source data. To integrate the source data into the cloud, you must store the sample data as a CSV file, upload the file to OBS, and use DataArts Studio DataArts Migration to migrate the data to other cloud services.

Step 1 Create a CSV file (UTF-8 without BOM) named **2017_Yellow_Taxi_Trip_Data.csv**, copy the sample data provided in the subsequent section to the CSV file, and save the file.

To generate a CSV file in Windows, you can perform the following steps:

1. Use a text editor (for example, Notepad) to create a .txt document and copy the sample data to the document. Then check the total number of rows and check whether the data of rows is correctly separated. (If the sample data is copied from a PDF document, the data in a single row will be wrapped if the data is too long. In this case, you must manually adjust the data to ensure that it is in a single row.)
2. Choose **File** > **Save as**. In the displayed dialog box, set **Save as type** to **All files (*.*)**, enter the file name with the .csv suffix for **File name**, and select the UTF-8 encoding format (without BOM) to save the file in CSV format.

Step 2 Upload the CSV file to OBS.

1. Log in to the management console and choose **Storage** > **Object Storage Service** to access the OBS console.
2. Click **Create Bucket** and set parameters as prompted to create an OBS bucket named **fast-demo**.

NOTE

To ensure network connectivity, select the same region for OBS bucket as that for the DataArts Studio instance. If an enterprise project is required, select the enterprise project that is the same as that of the DataArts Studio instance.

For details about how to create a bucket on the OBS console, see **Creating a Bucket** in *Object Storage Service Console Operation Guide*.

3. Upload data to OBS bucket **fast-demo**.

   For details about how to upload a file on the OBS console, see **Uploading a File** in *Object Storage Service Console Operation Guide*.

----End

The following shows partial sample data:

```
VendorID,tpep_pickup_datetime,tpep_dropoff_datetime,passenger_count,trip_distance,RatecodeID,store_and_
fwd_flag,PULocationID,DOLocationID,payment_type,fare_amount,extra,mta_tax,tip_amount,tolls_amount,im
provement_surcharge,total_amount
2,02/14/2017 04:08:11 PM,02/14/2017 04:21:53 PM,1,0.91,1,N,237,163,2,9.5,1,0.5,0,0,0.3,11.3
2,02/14/2017 04:08:11 PM,02/14/2017 04:19:29 PM,2,1.03,1,N,237,229,1,8.5,1,0.5,2.06,0,0.3,12.36
1,02/14/2017 04:08:12 PM,02/14/2017 04:19:44 PM,1,1.6,1,N,186,163,2,9,1,0.5,0,0,0.3,10.8
1,02/14/2017 04:08:12 PM,02/14/2017 04:19:15 PM,1,1.2,1,N,48,48,2,8.5,1,0.5,0,0,0.3,10.3
2,02/14/2017 04:08:12 PM,02/14/2017 04:13:38 PM,5,0.61,1,N,161,162,1,5.5,1,0.5,2.19,0,0.3,9.49
2,02/14/2017 04:08:12 PM,02/14/2017 05:35:11 PM,1,19.31,2,N,152,132,1,52,4.5,0.5,12.57,5.54,0.3,75.41
1,02/14/2017 04:08:13 PM,02/14/2017 04:20:53 PM,1,1.9,1,N,236,143,1,10.5,1,0.5,1.85,0,0.3,14.15
2,02/14/2017 04:08:13 PM,02/14/2017 04:15:54 PM,1,0.61,1,N,48,164,1,6.5,1,0.5,1.66,0,0.3,9.96
2,02/14/2017 04:08:13 PM,02/14/2017 04:41:40 PM,1,6.04,1,N,244,262,1,25,1,0.5,6.7,0,0.3,33.5
2,02/14/2017 04:08:13 PM,02/14/2017 04:17:31 PM,1,1.39,1,N,170,234,1,8,1,0.5,1,0,0.3,10.8
2,02/14/2017 04:08:14 PM,02/14/2017 04:54:11 PM,2,10.12,1,N,140,189,1,37.5,1,0.5,7,0,0.3,46.3
2,02/14/2017 04:08:14 PM,02/14/2017 04:13:56 PM,1,0.71,1,N,179,7,2,5.5,1,0.5,0,0,0.3,7.3
2,02/14/2017 04:08:14 PM,02/14/2017 05:04:24 PM,1,18.1,2,N,263,132,1,52,4.5,0.5,15.71,5.54,0.3,78.55
2,02/14/2017 04:08:14 PM,02/14/2017 04:08:47 PM,1,0.02,1,N,231,231,2,2.5,1,0.5,0,0,0.3,4.3
2,02/14/2017 04:08:15 PM,02/14/2017 04:18:13 PM,1,1.34,1,N,100,162,1,8,1,0.5,1.2,0,0.3,11
1,02/14/2017 04:08:16 PM,02/14/2017 04:19:01 PM,1,1.8,1,N,239,151,1,9,1,0.5,2.15,0,0.3,12.95
2,02/14/2017 04:08:16 PM,02/14/2017 04:15:57 PM,1,1.06,1,N,68,170,1,6.5,1,0.5,1,0,0.3,9.3
2,02/14/2017 04:08:16 PM,02/14/2017 04:20:08 PM,2,1.5,1,N,161,142,1,9,1,0.5,2.16,0,0.3,12.96
2,02/14/2017 04:08:16 PM,02/14/2017 04:11:56 PM,1,0.62,1,N,87,88,2,4.5,1,0.5,0,0,0.3,6.3
2,02/14/2017 04:08:16 PM,02/14/2017 04:13:20 PM,1,0.88,1,N,262,236,2,5.5,1,0.5,0,0,0.3,7.3
```

The following table lists the taxi trip data:

Table 4-3 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.

No.	Field Name	Field Description
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.
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.

Preparing a Data Lake

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.

DataArts Studio can integrate cloud services such as DWS, DLI, and MRS Hive, as well as conventional databases such as MySQLOracle. For details, see **Data Sources**.

In this example, MapReduce Service (MRS) Hive is used as the data lake base of DataArts Studio. You need to create an MRS security cluster (with Kerberos authentication enabled) because DataArts Security cannot be interconnected with common MRS clusters. For details, see **Buying a Cluster**.

To ensure that the MRS cluster can communicate with the DataArts Studio instance, the MRS cluster must meet the following requirements:

- The MRS cluster must contain a Hive component.
- If you want to enable automatic generation of quality jobs based on the data standards in DataArts Studio DataArts Architecture, ensure that the MRS cluster version is 2.0.3 or later and that the cluster contains Hive and Spark components and at least four nodes. In this example, this function is required.

  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

  During the creation of an MRS cluster, a security group is automatically created. You are advised to create an MRS security cluster first and then buy a DataArts Studio instance, selecting the same VPC and subnet as the MRS cluster and the security group (named in **mrs**_*Cluster name_Random character* format) that is automatically created. This ensures that the DataArts Studio instance can communicate with the MRS cluster by default.

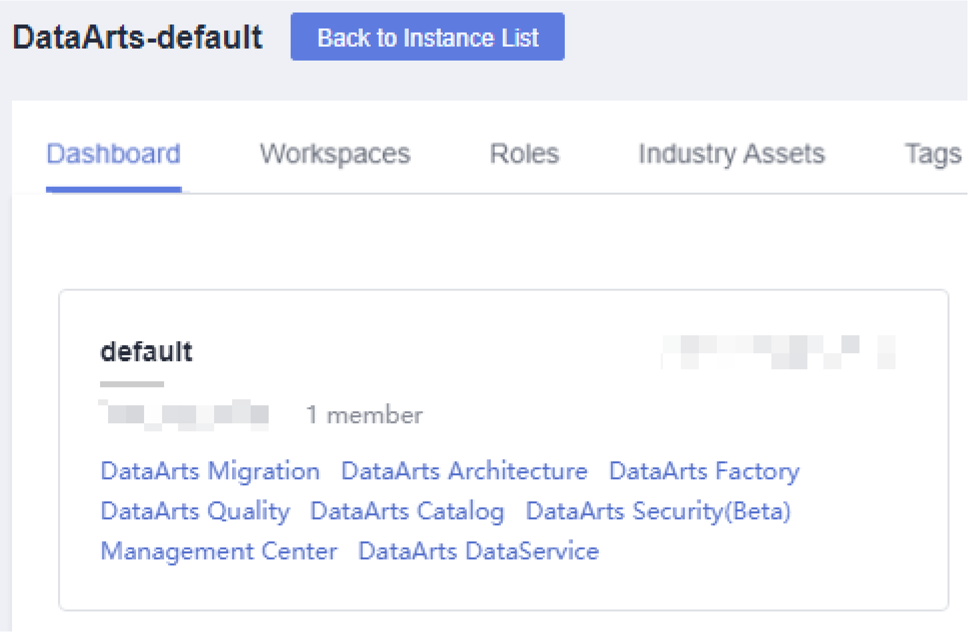
  If you already have a DataArts Studio instance before creating an MRS cluster, you need to choose **Access Control** > **Security Groups** on the VPC console and add a rule to allow inbound traffic to the security group (named in **mrs**_*Cluster name_Random character* format) created by the MRS cluster. For details, see **Configuring Security Group Rules**.

Creating a Data Connection on Management Center

After the data lake is prepared, create a data connection on Management Center to connect to the cloud service that functions as the data lake.

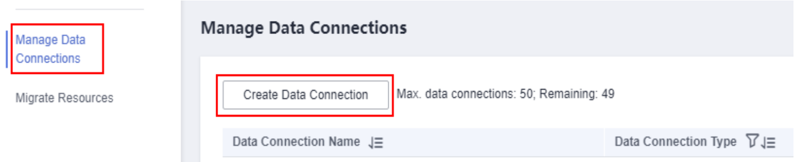
Step 1 On the DataArts Studio console, locate a workspace and click **Management Center**.

Figure 4-2 Management Center



Step 2 On the displayed **Manage Data Connections** page, click **Create Data Connection**.

Figure 4-3 Creating a data connection



Max. data connections: 50; Remaining: 49

Step 3 In the dialog box displayed, set data connection parameters and click **OK**.

The following part describes how to create an MRS Hive connection. See **Figure 4-4** for details.

- Data connection type: **MRS Hive**
- **Name**: Enter **mrs_hive_link**.
- **Tag**: Enter a new tag name or select an existing tag from the drop-down list box. This parameter is optional.
- **Cluster Name**: Select an existing MRS cluster.
- **Username**: Enter the Kerberos authentication user. In an MRS policy, user **admin** is the default management user and cannot be used as the authentication user of the cluster that uses Kerberos authentication. Therefore, to create a connection for an MRS cluster that uses Kerberos authentication, perform the following operations:

  a. Log in to the MRS Manager as user **admin**.

b. 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.

c. 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.

d. Log in to MRS Manager as the new user and change the initial password. Otherwise, the connection fails to be created.

e. Synchronize IAM users.

   i. Log in to the MRS console.

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

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

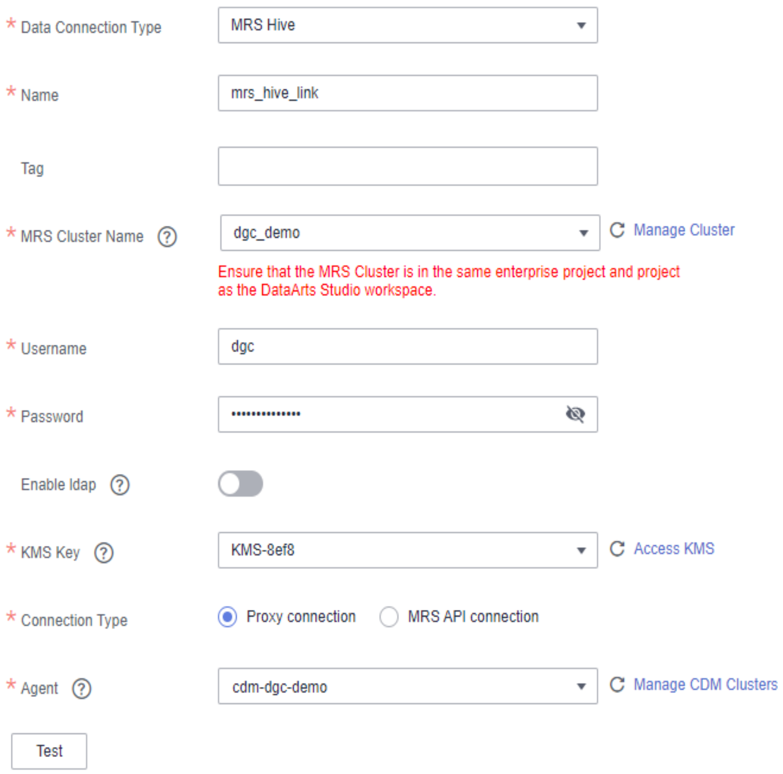
   NOTE

   - 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.

- **Password**: Enter the password of the Kerberos authentication user.
- **KMS Key**: Select a KMS key and use it to encrypt sensitive data. If no KMS key is available, click **Access KMS** to go to the KMS console and create one.
- **Connection Type**: Select **Proxy connection**.
- **Agent**: Select a DataArts Migration cluster as the connection agent. The DataArts Migration cluster and MRS cluster must be in the same region, AZ, VPC, and subnet, and the security group rule must allow communication between the two clusters. In this example, select the DataArts Migration cluster that is automatically created during DataArts Studio instance creation.

To connect to an MRS 2.*x* cluster, select the DataArts Migration cluster of the 2.*x* version as the agent.

Figure 4-4 Creating an MRS Hive data connection



* Data Connection Type	MRS Hive
* Name	mrs_hive_link
Tag	
* MRS Cluster Name	dgc_demo Ensure that the MRS Cluster is in the same enterprise project and project as the DataArts Studio workspace.
* Username	dgc
* Password	••••••••••••••
Enable ldap	
* KMS Key	KMS-8ef8
* Connection Type	● Proxy connection ○ MRS API connection
* Agent	cdm-dgc-demo

----End

Creating a Database

According to the implementation process of data lake governance, you are advised to create a database for each of the layers (SDI layer, DWI layer, DWR layer, and DM layer) in the data lake to implement hierarchical sharding. Data sharding is a concept involved in DataArts Architecture.

- **Source Data Integration (SDI)** copies data from the source system.
- **Data Warehouse Integration (DWI)** integrates and cleanses data from multiple source systems, and builds ER models based on the third normal form (3NF).
- **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 and displayed.

Generally, create a database in the data lake service.

In this example, you can use either of the following methods to create a database in MRS Hive:

- You can create a database on the DataArts Factory module of DataArts Studio. For details, see **Creating a Database**.
- You can also develop and execute a SQL script for creating a database using the DataArts Studio DataArts Factory module or on the MRS client, and then use the script to create a database. For details about how to develop a script in DataArts Factory, see **Developing an SQL Script**. For details about how to develop a script using the MRS Client, see **Using Hive from Scratch**. Run the following Hive SQL commands to create a database:

```
-- Create an SDI layer database.
CREATE DATABASE demo_sdi_db;

-- Create a DWI layer database.
CREATE DATABASE demo_dwi_db;

-- Create a DWR layer database.
CREATE DATABASE demo_dwr_db;

-- Create a DM layer database.
CREATE DATABASE demo_dm_db;
```

Creating Tables

Based on sample data, create a source table to store raw data. To migrate data from a file to a database, you must create a destination table in advance. In this example, the data source is a CSV file on OBS instead of a database. When you use DataArts Studio DataArts Migration to migrate data to the cloud, the destination table cannot be automatically created. Therefore, you must create a table on the destination (MRS).

NOTE

During data migration using DataArts Studio, a destination table can be automatically created for migration from relational databases to Hive and between relational databases. In this case, you do not need to create a table in the destination database in advance.

Run the following SQL statements to create a source table in the **demo_sdi_db** database to store raw data.

In this example, you can use either of the following methods to create a table in MRS Hive:

- You can create a table on the DataArts Studio DataArts Factory module. For details, see **Creating a Table**.
- You can also develop and execute a SQL script for creating a table using the DataArts Studio DataArts Factory module or on the MRS client, and then use the script to create a table. For details about how to develop a script in DataArts Factory, see **Developing an SQL Script**. For details about how to develop a script using the MRS Client, see **Using Hive from Scratch**. The following is an example Hive SQL command used to create a raw table in the **demo_sdi_db** database.

```
DROP TABLE IF EXISTS `sdi_taxi_trip_data`;

CREATE TABLE demo_sdi_db.`sdi_taxi_trip_data` (
  `VendorID` BIGINT COMMENT '',
  `tpep_pickup_datetime` TIMESTAMP COMMENT '',
  `tpep_dropoff_datetime` TIMESTAMP COMMENT '',
  `passenger_count` BIGINT COMMENT '',
  `trip_distance` DECIMAL(10,2) COMMENT '',
  `ratecodeid` BIGINT COMMENT '',
  `store_fwd_flag` STRING COMMENT '',
  `PULocationID` STRING COMMENT '',
  `DOLocationID` STRING COMMENT '',
  `payment_type` BIGINT COMMENT '',
  `fare_amount` DECIMAL(10,2) COMMENT '',
  `extra` DECIMAL(10,2) COMMENT '',
  `mta_tax` DECIMAL(10,2) COMMENT '',
  `tip_amount` DECIMAL(10,2) COMMENT '',
  `tolls_amount` DECIMAL(10,2) COMMENT '',
  `improvement_surcharge` DECIMAL(10,2) COMMENT '',
  `total_amount` DECIMAL(10,2) COMMENT ''
);
```

4.4 Step 3: DataArts Migration

This topic describes how to use DataArts Studio DataArts Migration to migrate source data to the cloud in batches.

Creating a Cluster

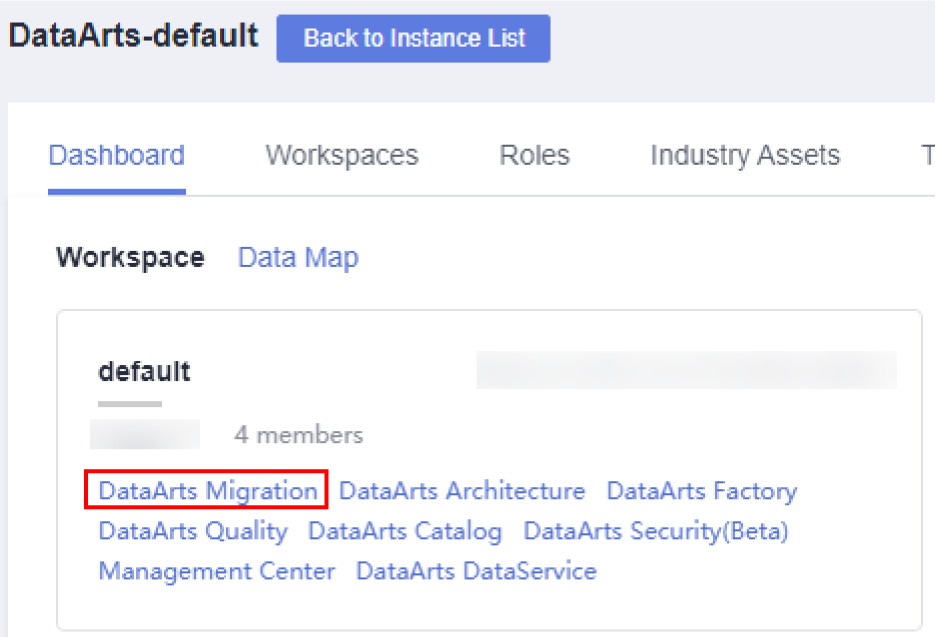
CDM clusters can migrate data to the cloud and integrate data into the data lake. It provides wizard-based configuration and management and can integrate data from a single table or an entire database incrementally or periodically. The DataArts Studio basic package contains a CDM cluster. If the cluster cannot meet your requirements, you can buy a DataArts Migration incremental package.

For details about how to buy DataArts Studio incremental packages, see **Buying a DataArts Studio Incremental Package**.

Creating Source and Destination Links for Data Migration

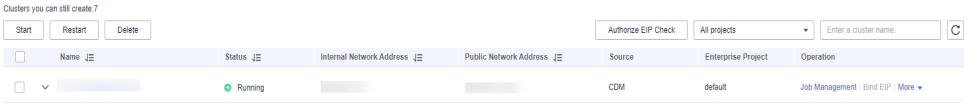
Step 1 Log in to the DataArts Studio console. Locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Migration**.

Figure 4-5 DataArts Migration



Step 2 In the left navigation pane, choose **Cluster Management**. In the cluster list, locate the required cluster and click **Job Management**.

Figure 4-6 Cluster management

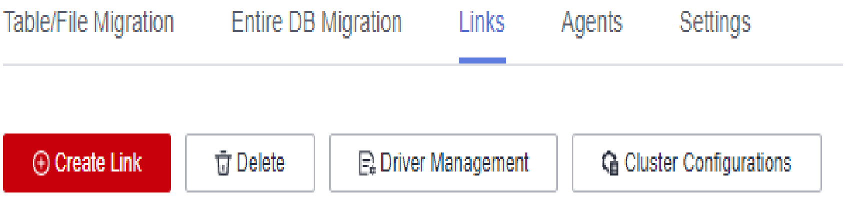


Clusters you can still create:7

Name	Status	Internal Network Address	Public Network Address	Source	Enterprise Project	Operation
[redacted]	Running	[redacted]	[redacted]	CDM	default	Job Management | Bind EIP | More

Step 3 On the **Job Management** page, click **Links**.

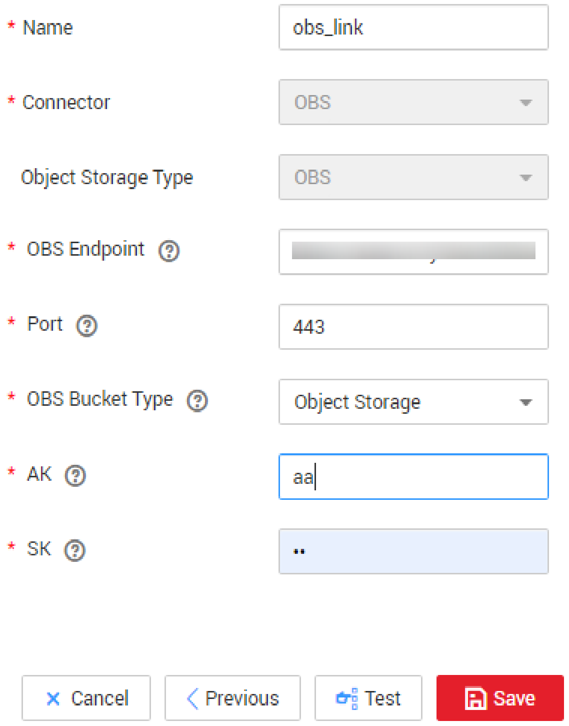
Figure 4-7 Links



Step 4 Create two links, one connecting to OBS to read source data stored on OBS, and the other connecting to MRS Hive to write data to the MRS Hive database.

Click **Create Link**. On the page displayed, select **Object Storage Service (OBS)** and click **Next**. Then, set the link parameters and click **Save**.

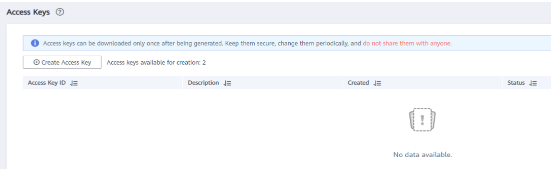
Figure 4-8 Creating an OBS link



* Name	obs_link
* Connector	OBS
Object Storage Type	OBS
* OBS Endpoint	[redacted]
* Port	443
* OBS Bucket Type	Object Storage
* AK	aa
* SK	••

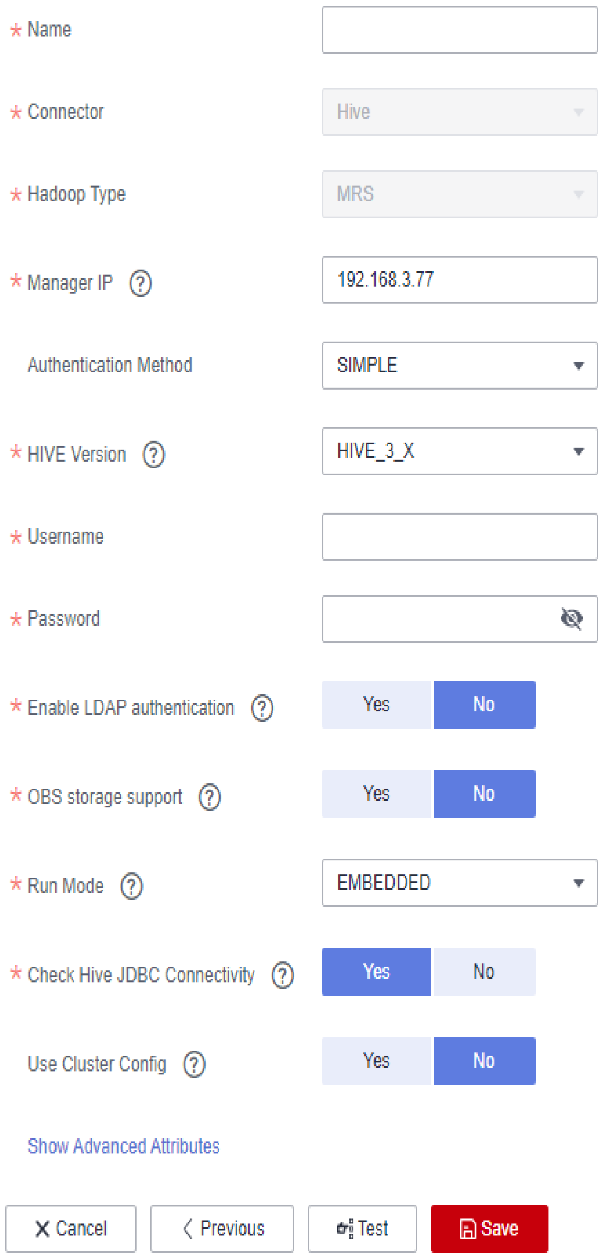
Table 4-4 Parameter description

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
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.	-
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 4-9**. **Figure 4-9** Clicking Create Access Key  Access Keys Access keys can be downloaded only once after being generated. Keep them secure, change them periodically, and do not share them with anyone. Create Access Key Access keys available for creation: 2 Access Key ID | Description | Created | Status No data available. 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		-

On the **Links** tab page, click **Create Link** again. On the page displayed, select **MRS Hive** and click **Next**. Then, set the link parameters and click **Save**.

Figure 4-10 Creating an MRS Hive link



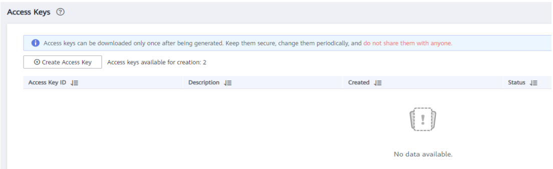
* Name	
* Connector	Hive
* Hadoop Type	MRS
* Manager IP	192.168.3.77
Authentication Method	SIMPLE
* HIVE Version	HIVE_3_X
* Username	
* Password	
* Enable LDAP authentication	Yes [✓ No]
* OBS storage support	Yes [✓ No]
* Run Mode	EMBEDDED
* Check Hive JDBC Connectivity	[✓ Yes] No
Use Cluster Config	Yes [✓ No]

Table 4-5 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 4-11**. **Figure 4-11** Clicking Create Access Key  Access Keys Access keys can be downloaded only once after being generated. Keep them secure, change them periodically, and do not share them with anyone. Create Access Key Access keys available for creation: 2 Access Key ID Description Created Status No data available. 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

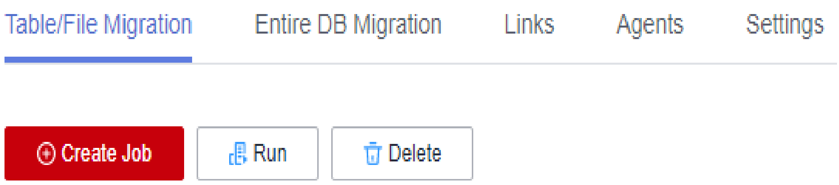
----End

Creating a Table/File Migration Job

Step 1 On the DataArts Migration console, click **Cluster Management** in the left navigation pane, locate the required cluster in the cluster list, and click **Job Management**.

Step 2 On the **Job Management** page, click **Table/File Migration** and click **Create Job**.

Figure 4-12 Table/File Migration



Step 3 Set job parameters:

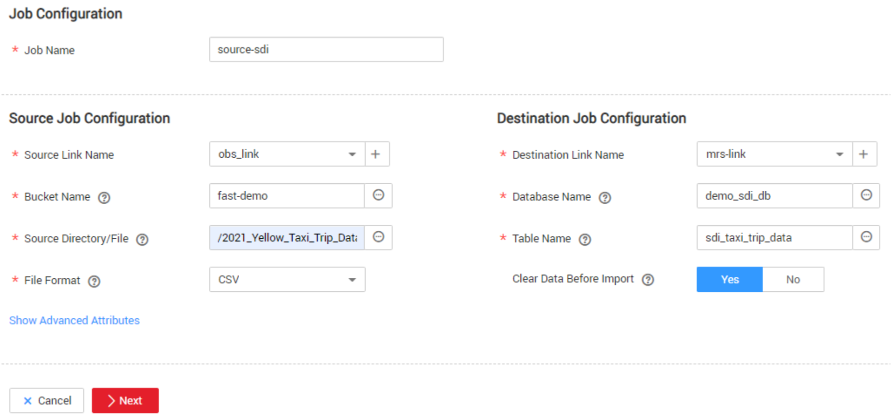
1. Configure the job name, source job parameters, and destination job parameters, and click **Next**. See **Figure 4-13**.
   - **Job Name**: **source-sdi**
   - **Source Job Configuration**
     - **Source Link Name**: **obs-link**
     - **Bucket Name**: **fast-demo**
     - **Source Directory/File**: **/2017_Yellow_Taxi_Trip_Data.csv**
     - **File Format**: **CSV**
     - **Show Advanced Attributes**: Click **Show Advanced Attributes**. The system provides default values for advanced attributes. Set parameters based on the actual data format.

       Pay attention to the settings of the following parameters based on the sample data format in **Preparing a Data Source**. For other parameters, retain the default values.
       - **Field Delimiter**: Retain the default value (**,**) in this example.
       - **First N Rows As Header**: Set this parameter to **Yes** because the first row is the title row in this example.
       - **The Number of Header Rows**: Enter **1**.
       - **Encode Type**: Retain the default value **UTF-8** in this example.
   - **Destination Job Configuration**
     - **Destination Link Name**: **mrs-link**
     - **Database Name**: **demo_sdi_db**
     - **Table Name**: **sdi_taxi_trip_data**
     - **Clear Data Before Import**

       NOTE

       In this example, **Clear Data Before Import** is set to **Yes**, indicating that data will be cleared before being imported each time a job is executed. In actual services, set this parameter based on the site requirements to prevent data loss.

Figure 4-13 Configuring basic job information



Job Configuration

* Job Name: source-sdi

Source Job Configuration

* Source Link Name: obs_link

* Bucket Name: fast-demo

* Source Directory/File: /2021_Yellow_Taxi_Trip_Dat

* File Format: CSV

Destination Job Configuration

* Destination Link Name: mrs-link

* Database Name: demo_sdi_db

* Table Name: sdi_taxi_trip_data

Clear Data Before Import: [✓ Yes] No

2. In the **Map Field** step, configure field mappings and the time format of date fields, as shown in **Figure 4-14**. After the configuration is complete, click **Next**.
   - **Field Mapping**: In this example, the field sequence in the destination table is the same as that of source data. Therefore, you do not need to adjust the field mapping sequence.

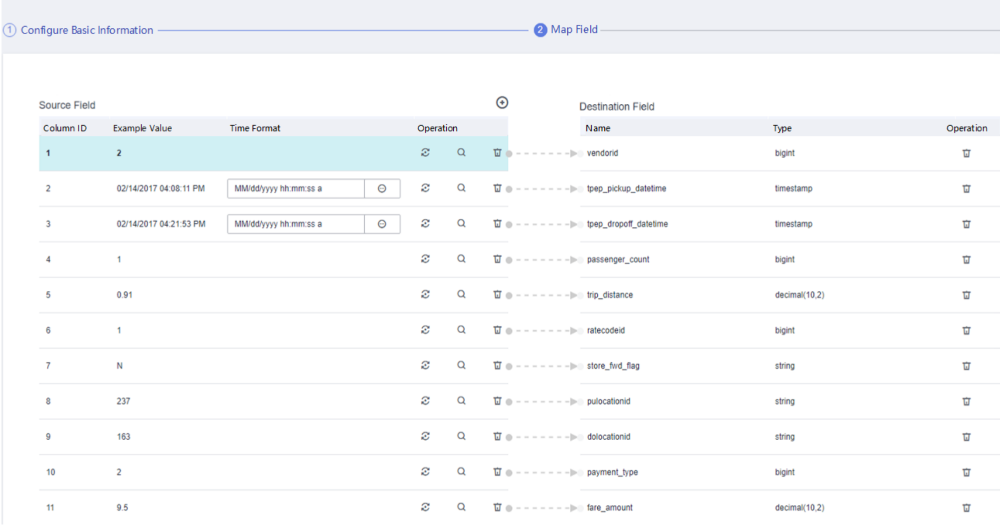
     If the field sequence in the destination table is different from that of source data, map the source fields one by one to the destination fields with the same meaning. Move the cursor to the start point of the arrow of a field. When the cursor is displayed as a plus sign (+), press and hold the mouse button, point the arrow to the destination field with the same meaning, and then release the button.
   - **Time Format**: The second and third fields in the sample data are time fields. The data format is 02/14/2017 04:08:11 PM. Therefore, set **Time Format** to **MM/dd/yyyy hh:mm:ss a** for these two fields. You can also manually enter this format in the text box.

     Select the time format based on the actual data format. For example:

     yyyy/MM/dd HH:mm:ss indicates that the time is converted to the 24-hour format, for example, 2019/08/18 15:35:45.

     yyyy/MM/dd hh:mm:ss a indicates that the time is converted to the 12-hour format, for example, 2019/06/27 03:24:21 PM.

Figure 4-14 Mapping fields



Source Field

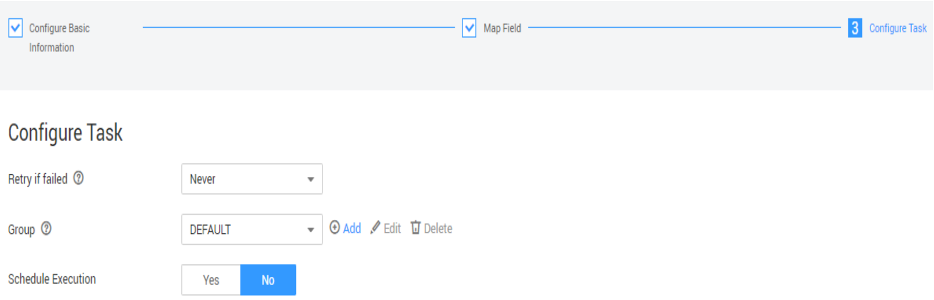
Column ID	Example Value	Time Format
1	2	
2	02/14/2017 04:08:11 PM	MM/dd/yyyy hh:mm:ss a
3	02/14/2017 04:21:53 PM	MM/dd/yyyy hh:mm:ss a
4	1	
5	0.91	
6	1	
7	N	
8	237	
9	163	
10	2	
11	9.5	

Destination Field

Name	Type
vendorid	bigint
tpep_pickup_datetime	timestamp
tpep_dropoff_datetime	timestamp
passenger_count	bigint
trip_distance	decimal(10,2)
ratecodeid	bigint
store_fwd_flag	string
pulocationid	string
dolocationid	string
payment_type	bigint
fare_amount	decimal(10,2)

3. Set **Retry if failed** and **Schedule Execution** of the task as required.

Figure 4-15 Configuring the task



Configure Task

Retry if failed: Never

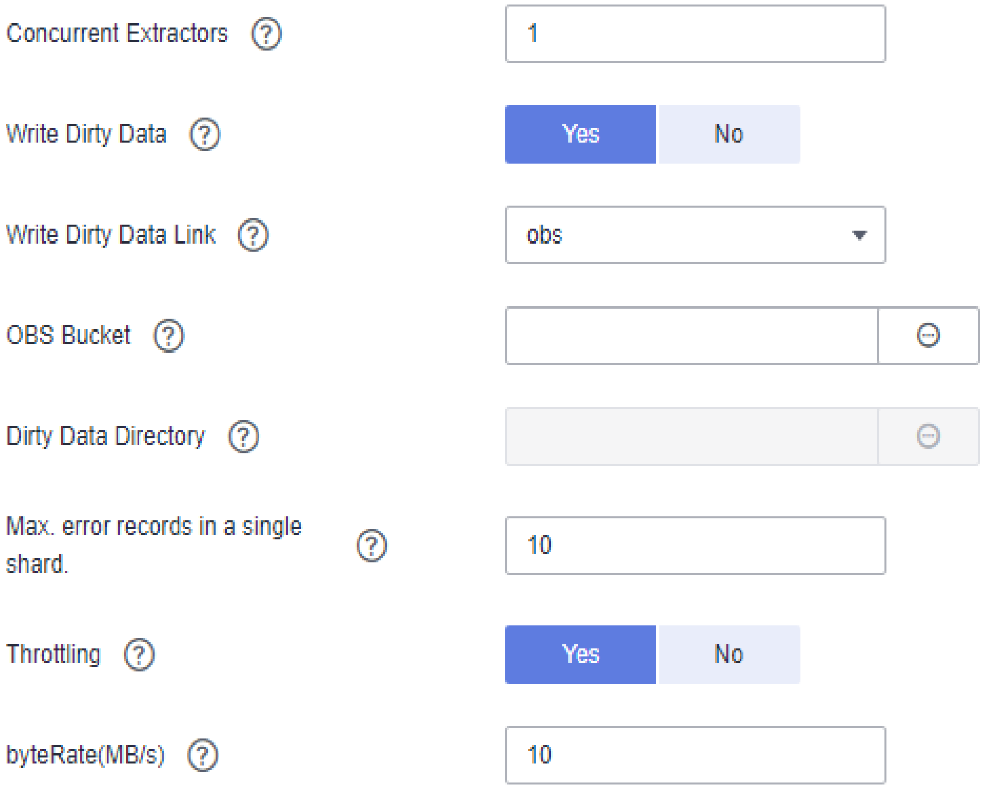
Group: DEFAULT

Schedule Execution: Yes [✓ No]

Click **Show Advanced Attributes** and set **Concurrent Extractors** and **Write Dirty Data**, as shown in **Figure 4-16**.

- **Concurrent Extractors**: Set this parameter based on the service volume. If the data source is of the file type and there are multiple files, you can increase the value of **Concurrent Extractors** to improve the extraction speed.
- **Write Dirty Data**: You are advised to set this parameter to **Yes** and set related parameters by referring to **Figure 4-16**. Dirty data refers to the data that does not match the fields at the migration destination. Such data will be recorded to a specified OBS bucket. After dirty data writing is configured, normal data will be written to the destination, and migration jobs will not be interrupted due to dirty data. In this example, set **OBS Bucket** to **fast-demo** created in **Preparing a Data Source**. Go to the OBS console, click **Create Folder** to create a directory, for example, **error-data**, in the **fast-demo** bucket, and configure the dirty data directory in **Figure 4-16** as the directory.

Figure 4-16 Advanced attributes

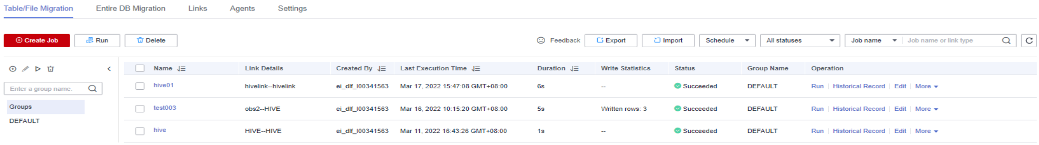


Concurrent Extractors ?	1
Write Dirty Data ?	[✓ Yes] No
Write Dirty Data Link ?	obs
OBS Bucket ?	
Dirty Data Directory ?	
Max. error records in a single shard. ?	10
Throttling ?	[✓ Yes] No
byteRate(MB/s) ?	10

Step 4 Click **Save**.

On the **Table/File Migration** tab page, you can view the created job in the job list.

Figure 4-17 Execution result of the migration task



Name	Link Details	Created By	Last Execution Time	Duration	Write Statistics	Status	Group Name	Operation
hive01	hivelink--hivelink	ei_dlf_l00341563	Mar 17, 2022 15:47:08 GMT+08:00	6s	--	Succeeded	DEFAULT	Run | Historical Record | Edit | More
test003	obs2--HIVE	ei_dlf_l00341563	Mar 16, 2022 10:15:20 GMT+08:00	5s	Written rows: 3	Succeeded	DEFAULT	Run | Historical Record | Edit | More
hive	HIVE--HIVE	ei_dlf_l00341563	Mar 11, 2022 16:43:26 GMT+08:00	1s	--	Succeeded	DEFAULT	Run | Historical Record | Edit | More

----End

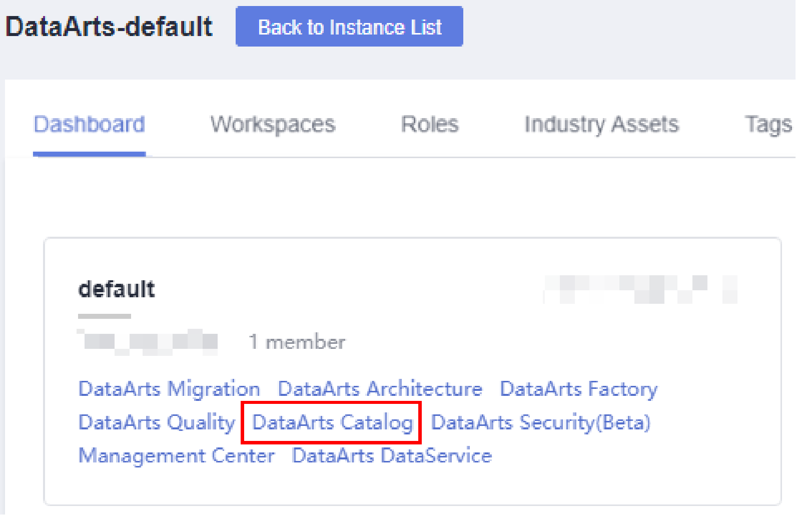
4.5 Step 4: Metadata Collection

To manage and monitor the raw data migrated to the cloud on DataArts Studio, you can use the DataArts Catalog module to collect and monitor the metadata at the Source Data Integration (SDI) layer.

Collecting and Monitoring Metadata

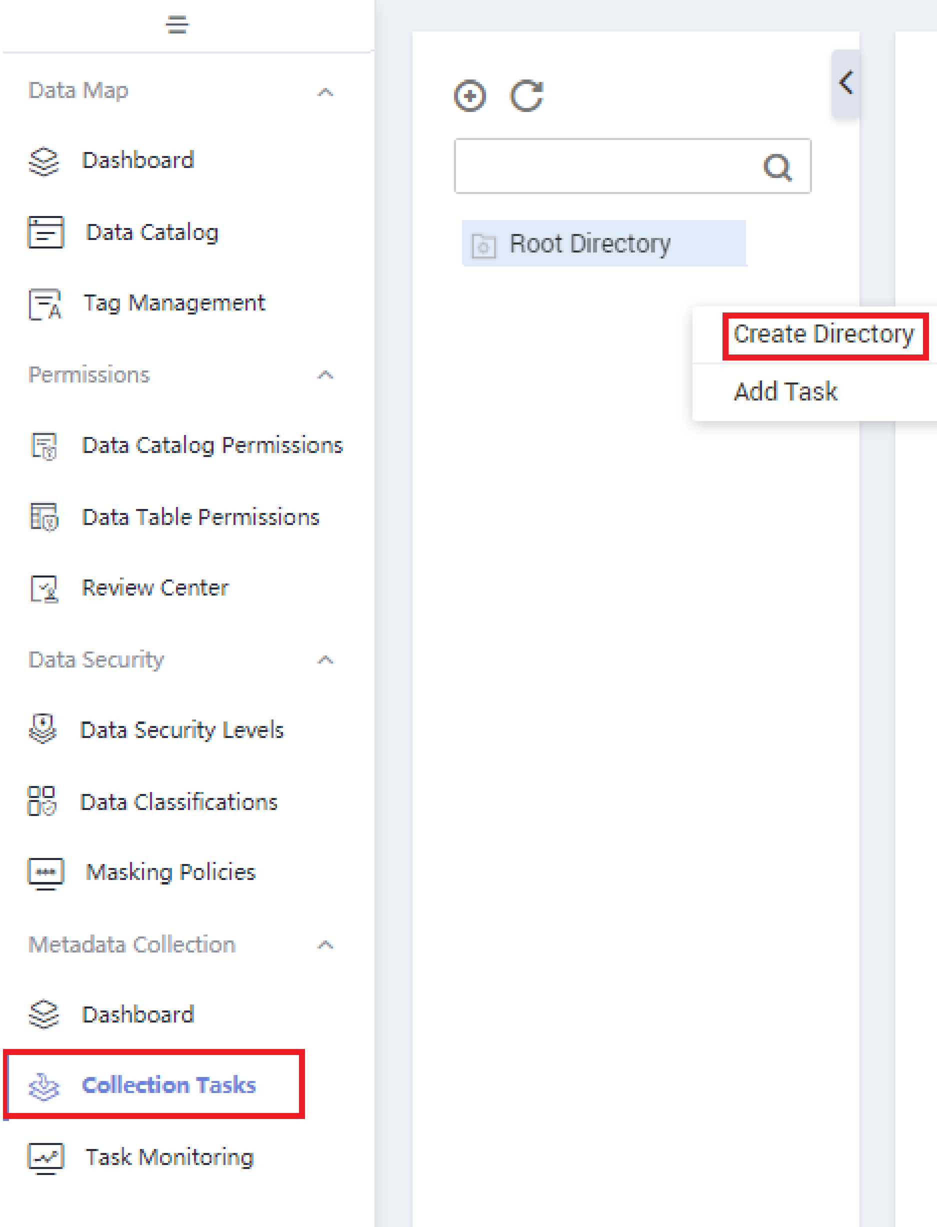
Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

Figure 4-18 DataArts Catalog



Step 2 Choose **Metadata Collection** > **Collection Tasks** in the left navigation pane, right-click a directory in the directory tree, and choose **Create Directory** from the shortcut menu. In the dialog box displayed, enter the directory name, for example, **transport**, select a parent directory, and click **OK**.

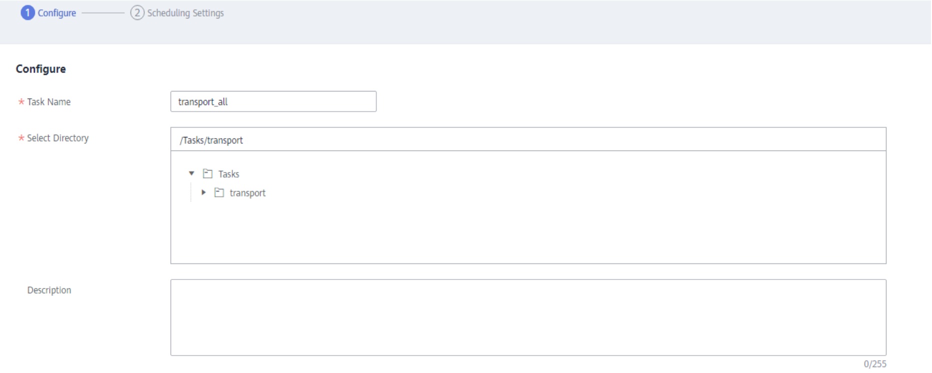
Figure 4-19 Collection Tasks



Step 3 Select the **transport** directory in the directory tree and click **Add Task**.

Step 4 Create a collection task named **transport_all**, configure parameters shown in the following figure, and click **Next**.

Figure 4-20 Creating a collection task (basic settings)



Configure

* Task Name: transport_all

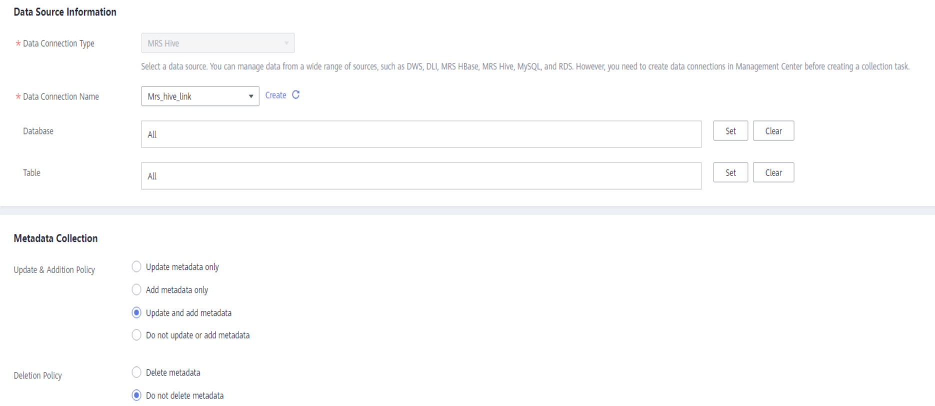
* Select Directory: /Tasks/transport

- Tasks
  - transport

Description:

0/255

Figure 4-21 Creating a metadata collection task



Data Source Information

* Data Connection Type: MRS Hive

Select a data source. You can manage data from a wide range of sources, such as DWS, DLI, MRS HBase, MRS Hive, MySQL, and RDS. However, you need to create data connections in Management Center before creating a collection task.

* Data Connection Name: Mrs_hive_link Create

Database: All Set Clear

Table: All Set Clear

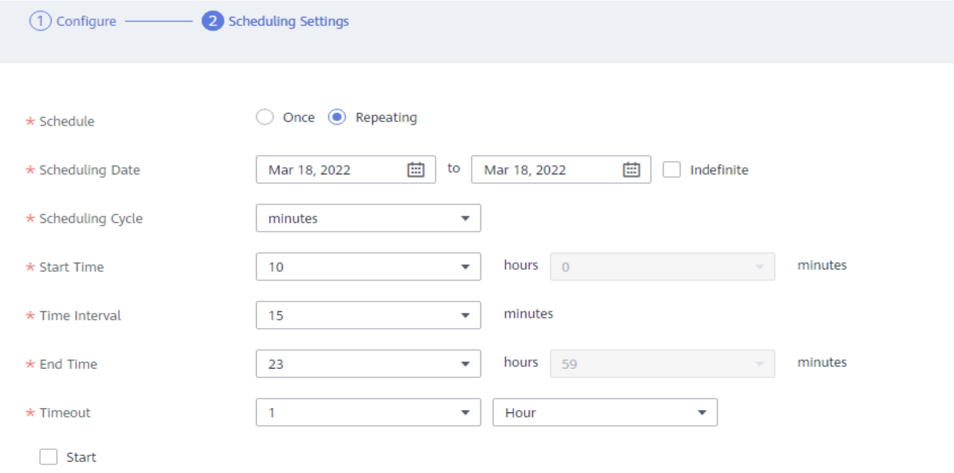
Metadata Collection

Update & Addition Policy:
○ Update metadata only
○ Add metadata only
● Update and add metadata
○ Do not update or add metadata

Deletion Policy:
○ Delete metadata
● Do not delete metadata

Step 5 Configure the scheduling mode and click **Submit**.

Figure 4-22 Configuring the scheduling mode



* Schedule: ○ Once ● Repeating

* Scheduling Date: Mar 18, 2022 to Mar 18, 2022 □ Indefinite

* Scheduling Cycle: minutes

* Start Time: 10 hours 0 minutes

* Time Interval: 15 minutes

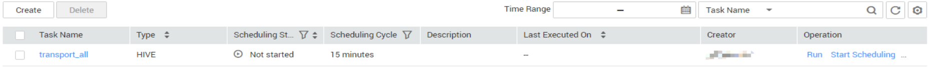
* End Time: 23 hours 59 minutes

* Timeout: 1 Hour

□ Start

Step 6 In the collection task list, locate the new collection task and click **Start Scheduling** in the row that contains the task.

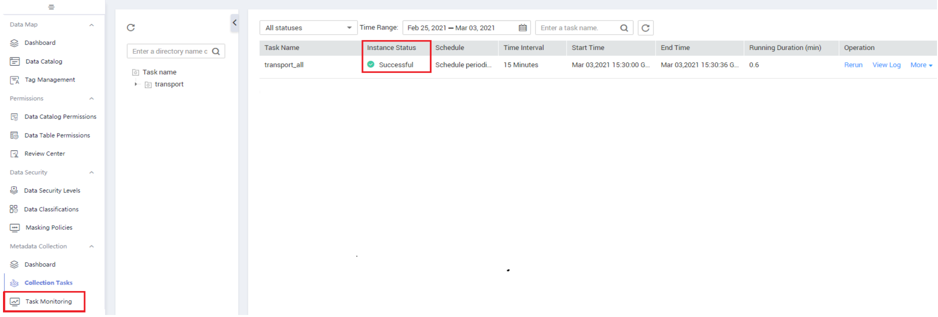
Figure 4-23 Starting scheduling



□	Task Name	Type	Scheduling St...	Scheduling Cycle	Description	Last Executed On	Creator	Operation
□	transport_all	HIVE	Not started	15 minutes		--	[redacted]	Run Start Scheduling ...

Step 7 Choose **Metadata Collection** > **Task Monitoring** in the left navigation pane, and check whether the collection task is successful.

Figure 4-24 Viewing a monitoring task

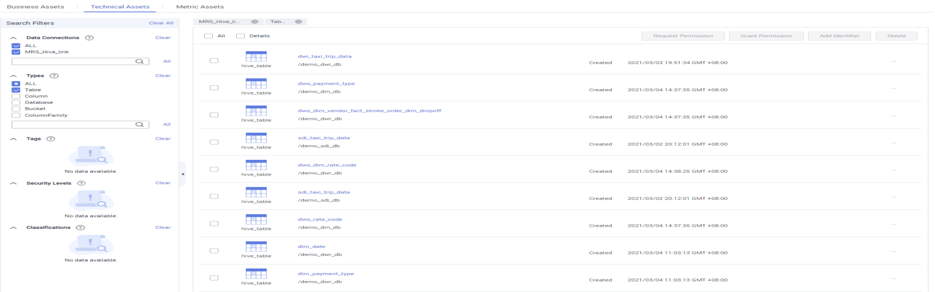


Time Range: Feb 25, 2021 — Mar 03, 2021

Task Name	Instance Status	Schedule	Time Interval	Start Time	End Time	Running Duration (min)	Operation
transport_all	Successful	Schedule periodi...	15 Minutes	Mar 03,2021 15:30:00 G...	Mar 03,2021 15:30:36 G...	0.6	Rerun View Log More

Step 8 After the collection task is successful, choose **Data Map** > **Data Catalog** in the left navigation pane, click the **Technical Assets** tab, and set filter criteria. For example, select **mrs_hive_link** for **Data Connections** and **Table** for **Types**. All tables that meet the filter criteria are displayed.

Figure 4-25 Technical assets



Data Connections
☑ ALL
☑ MRS_Hive_link

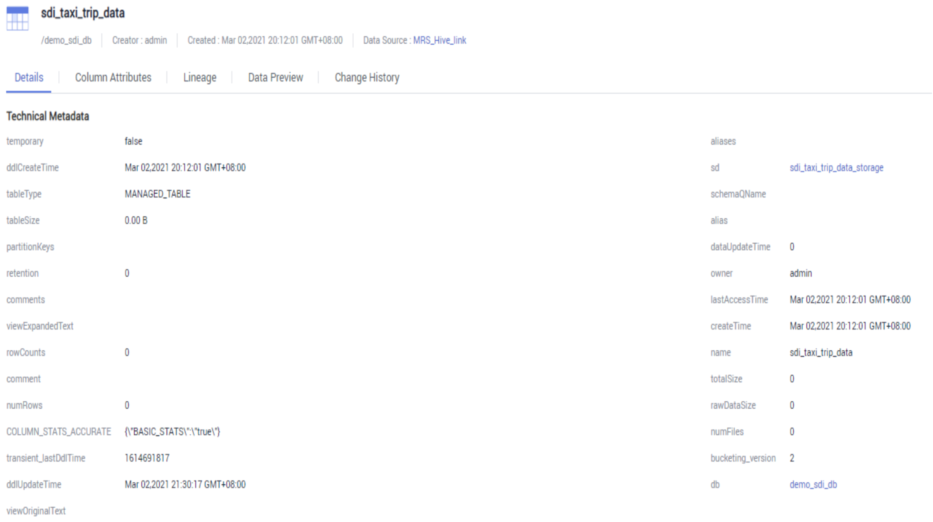
Types
☑ ALL
☑ Table
□ Column
□ Database
□ Bucket
□ ColumnFamily

□ All □ Details

□	hive_table	dwi_taxi_trip_data /demo_dwi_db	Created	2021/03/03 19:51:34 GMT +08:00
□	hive_table	dws_payment_type /demo_dm_db	Created	2021/03/04 14:37:35 GMT +08:00
□	hive_table	dws_dim_vendor_fact_stroke_order_dim_dropoff /demo_dwr_db	Created	2021/03/04 14:37:35 GMT +08:00
□	hive_table	sdi_taxi_trip_data /demo_sdi_db	Created	2021/03/02 20:12:01 GMT +08:00
□	hive_table	dws_dim_rate_code /demo_dwr_db	Created	2021/03/04 14:38:25 GMT +08:00
□	hive_table	sdi_taxi_trip_data /demo_sdi_db	Created	2021/03/02 20:12:01 GMT +08:00
□	hive_table	dws_rate_code /demo_dm_db	Created	2021/03/04 14:37:35 GMT +08:00
□	hive_table	dim_date /demo_dwr_db	Created	2021/03/04 11:03:13 GMT +08:00
□	hive_table	dim_payment_type /demo_dwr_db	Created	2021/03/04 11:03:13 GMT +08:00

Step 9 Click the target metadata name to view its details.

Figure 4-26 Metadata details



sdi_taxi_trip_data

/demo_sdi_db | Creator : admin | Created : Mar 02,2021 20:12:01 GMT+08:00 | Data Source : MRS_Hive_link

Technical Metadata

temporary	false	aliases	
ddlCreateTime	Mar 02,2021 20:12:01 GMT+08:00	sd	sdi_taxi_trip_data_storage
tableType	MANAGED_TABLE	schemaQName	
tableSize	0.00 B	alias	
partitionKeys		dataUpdateTime	0
retention	0	owner	admin
comments		lastAccessTime	Mar 02,2021 20:12:01 GMT+08:00
viewExpandedText		createTime	Mar 02,2021 20:12:01 GMT+08:00
rowCounts	0	name	sdi_taxi_trip_data
comment		totalSize	0
numRows	0	rawDataSize	0
COLUMN_STATS_ACCURATE	{\"BASIC_STATS\":\"true\"}	numFiles	0
transient_lastDdlTime	1614691817	bucketing_version	2
ddlUpdateTime	Mar 02,2021 21:30:17 GMT+08:00	db	demo_sdi_db
viewOriginalText			

----End

4.6 Step 5: DataArts Architecture

Use DataArts Studio 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.

The recommended data layers of DataArts Studio DataArts Architecture are as follows:

- **Source Data Integration (SDI)** copies data from the source system.
- **Data Warehouse Integration (DWI)** integrates and cleanses data from multiple source systems, and builds ER models based on the third normal form (3NF).
- **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 and displayed.

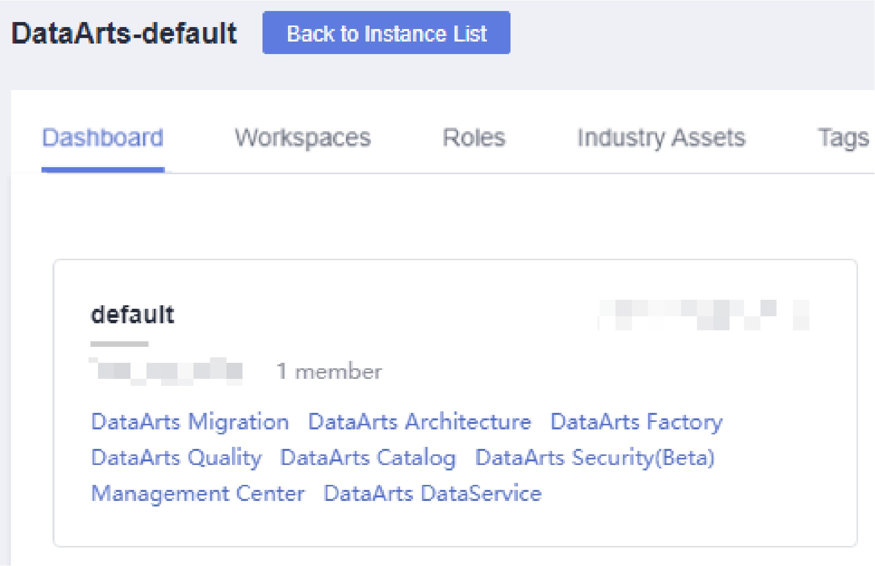
This topic describes how to use the DataArts Studio DataArts Architecture module to design models.

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**.

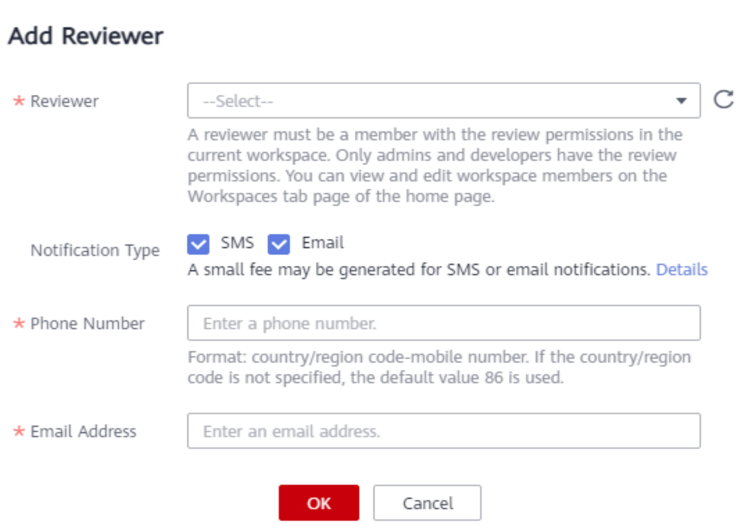
Figure 4-27 DataArts Architecture



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 4-28 Adding a reviewer



Add Reviewer

* Reviewer: --Select--

A reviewer must be a member with the review permissions in the current workspace. Only admins and developers have the review permissions. You can view and edit workspace members on the Workspaces tab page of the home page.

Notification Type: ☑ SMS ☑ Email

A small fee may be generated for SMS or email notifications. Details

* Phone Number: Enter a phone number.

Format: country/region code-mobile number. If the country/region code is not specified, the default value 86 is used.

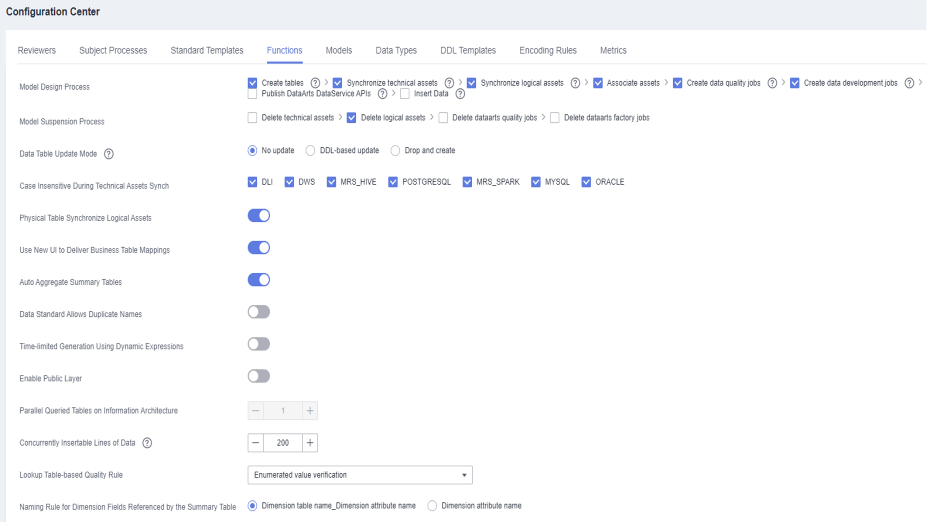
* Email Address: Enter an email address.

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 4-29 Functions



Configuration Center

Reviewers | Subject Processes | Standard Templates | Functions | Models | Data Types | DDL Templates | Encoding Rules | Metrics

Model Design Process: ☑ Create tables > ☑ Synchronize technical assets > ☑ Synchronize logical assets > ☑ Associate assets > ☑ Create data quality jobs > ☑ Create data development jobs > □ Publish DataArts DataService APIs > □ Insert Data

Model Suspension Process: □ Delete technical assets > ☑ Delete logical assets > □ Delete dataarts quality jobs > □ Delete dataarts factory jobs

Data Table Update Mode: ● No update ○ DDL-based update ○ Drop and create

Case Insensitive During Technical Assets Synch: ☑ DLI ☑ DWS ☑ MRS_HIVE ☑ POSTGRESQL ☑ MRS_SPARK ☑ MYSQL ☑ ORACLE

Physical Table Synchronize Logical Assets: ☑

Use New UI to Deliver Business Table Mappings: ☑

Auto Aggregate Summary Tables: ☑

Data Standard Allows Duplicate Names: □

Time-limited Generation Using Dynamic Expressions: □

Enable Public Layer: □

Parallel Queried Tables on Information Architecture: 1

Concurrently Insertable Lines of Data: 200

Lookup Table-based Quality Rule: Enumerated value verification

Naming Rule for Dimension Fields Referenced by the Summary Table: ● Dimension table name_Dimension attribute name ○ Dimension attribute name

3. Click **OK**.

Designing a Subject

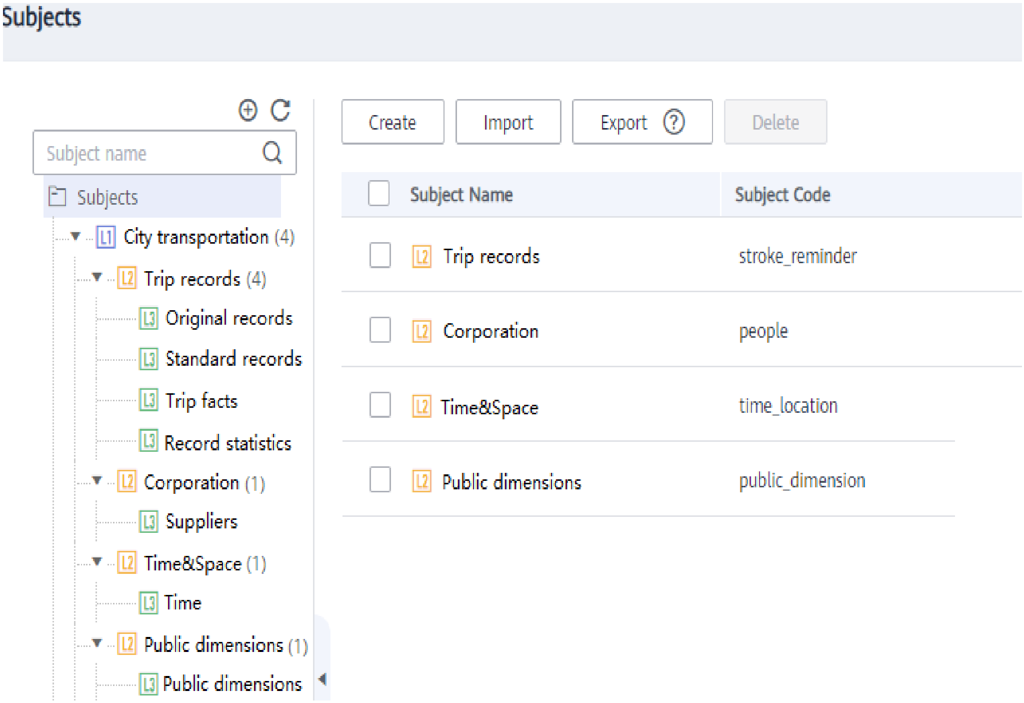
This section uses the subjects listed in **Table 4-6** as an example.

- 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 4-6 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 4-30 Designing a subject



Subjects

□ Subject Name	Subject Code
□ [L2] Trip records	stroke_reminder
□ [L2] Corporation	people
□ [L2] Time&Space	time_location
□ [L2] Public dimensions	public_dimension

Procedure

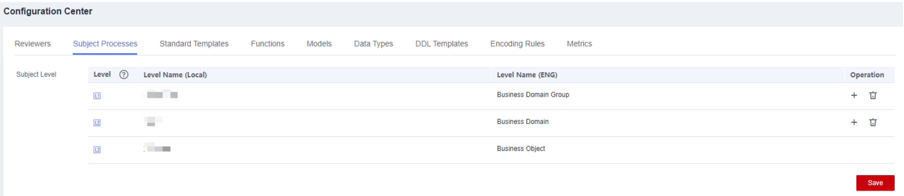
Step 1 Log in to the DataArts Studio console. Locate the created DataArts Studio instance and click **Access**.

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 4-31 Configuring the subject levels

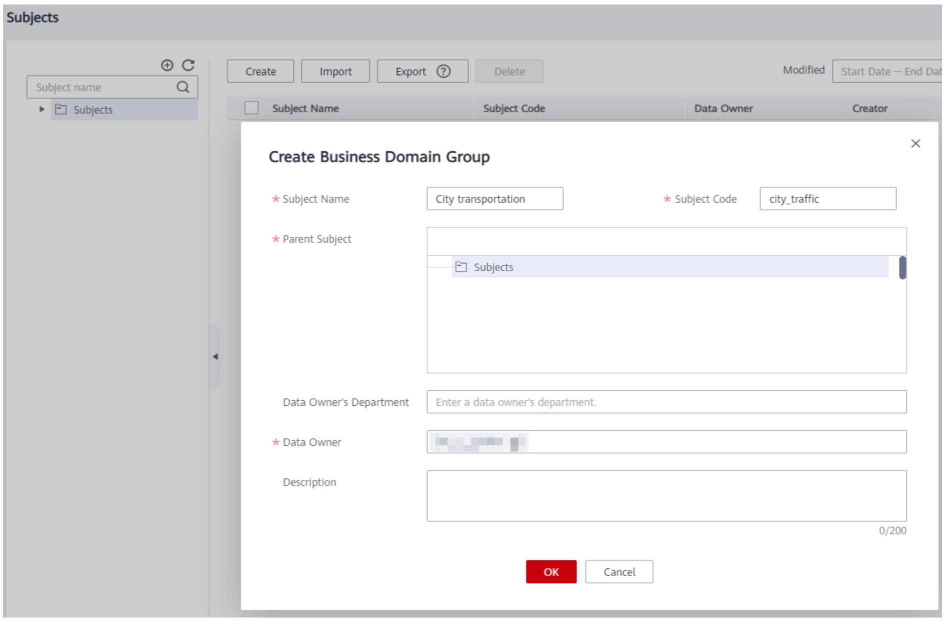


Configuration Center

Level	Level Name (Local)	Level Name (ENG)	Operation
L1		Business Domain Group	+ 🗑
L2		Business Domain	+ 🗑
L3		Business Object	

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 4-32 Creating an L1 subject



Create Business Domain Group

* Subject Name: City transportation

* Subject Code: city_traffic

* Parent Subject: Subjects

Data Owner's Department: Enter a data owner's department.

* Data Owner: [redacted]

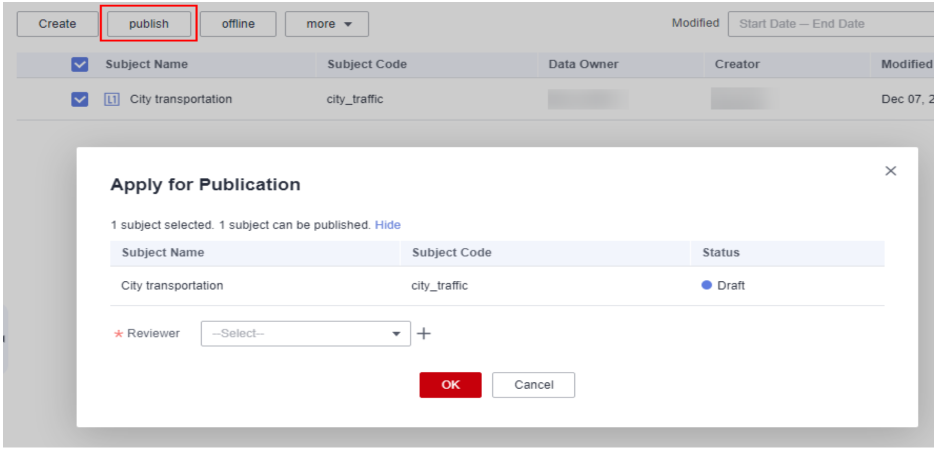
Description:

0/200

In the dialog box displayed, set the parameters as shown in **Figure 4-32** 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 4-33 Publishing a subject area group



☑	Subject Name	Subject Code	Data Owner	Creator	Modified
☑	City transportation	city_traffic	[redacted]	[redacted]	Dec 07, 2

Apply for Publication

1 subject selected. 1 subject can be published. Hide

Subject Name	Subject Code	Status
City transportation	city_traffic	● Draft

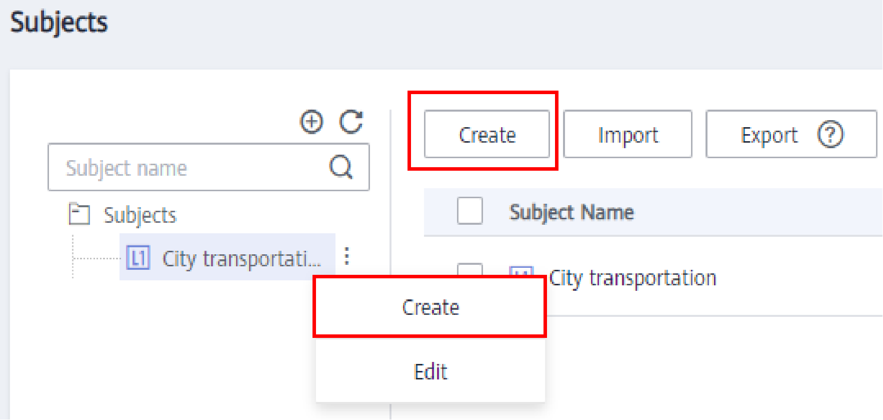
* Reviewer: --Select--

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.

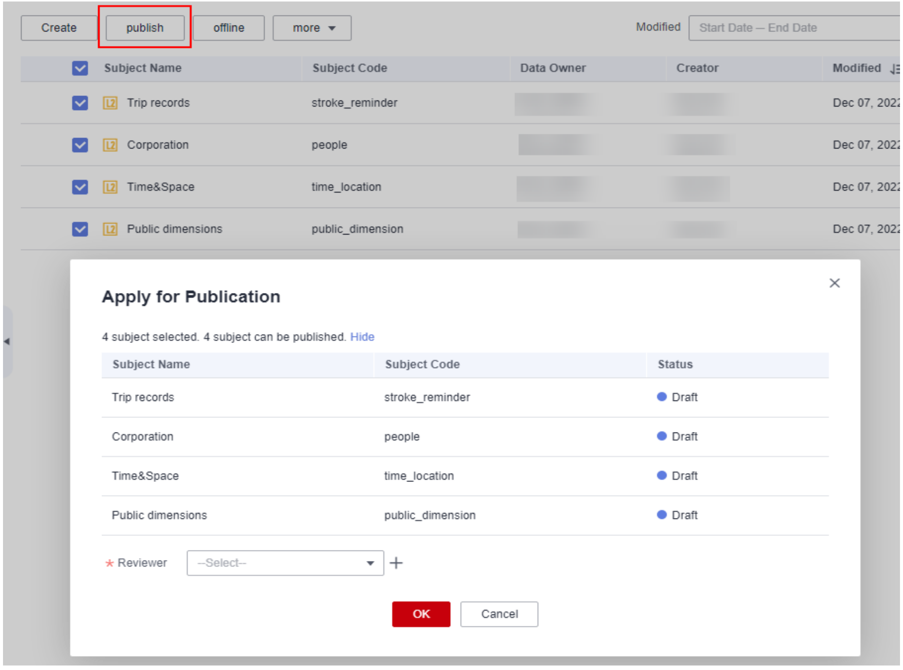
Figure 4-34 Creating an L2 subject



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 4-6**, 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 4-35 Publishing a subject area



☑ Subject Name	Subject Code	Data Owner	Creator	Modified
☑ Trip records	stroke_reminder	[redacted]	[redacted]	Dec 07, 202
☑ Corporation	people	[redacted]	[redacted]	Dec 07, 202
☑ Time&Space	time_location	[redacted]	[redacted]	Dec 07, 202
☑ Public dimensions	public_dimension	[redacted]	[redacted]	Dec 07, 202

Apply for Publication

4 subject selected. 4 subject can be published. Hide

Subject Name	Subject Code	Status
Trip records	stroke_reminder	● Draft
Corporation	people	● Draft
Time&Space	time_location	● Draft
Public dimensions	public_dimension	● Draft

* Reviewer --Select--

Step 7 Create business objects.

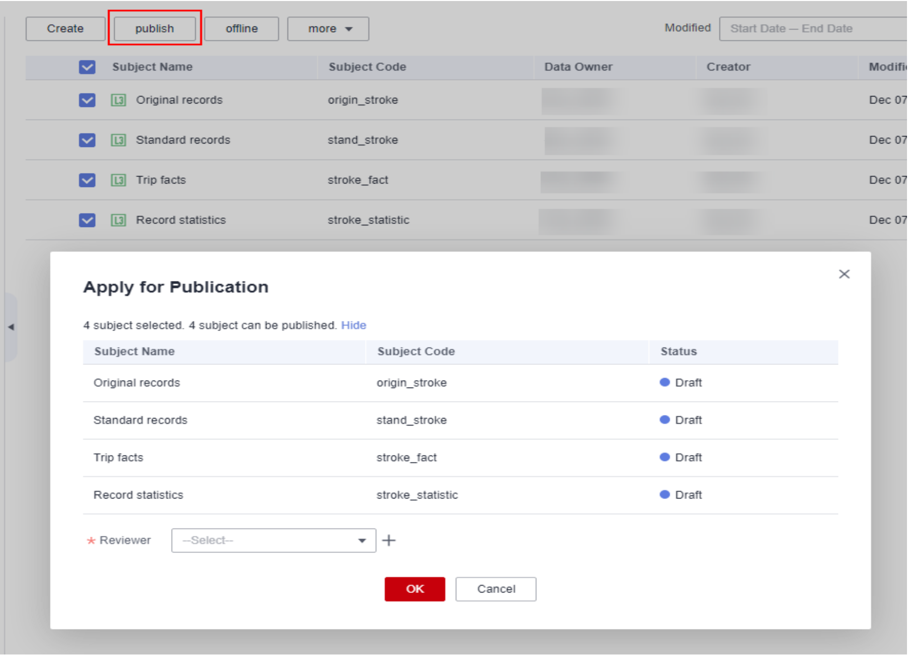
- Under **Trip records**, create four business objects: **Original records**, **Standard records**, **Trip facts**, and **Record statistics**.
- Under **Corporation**, create one business object: **Suppliers**.

- 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 4-6**, 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 4-36 Publishing a business object



Subject Name	Subject Code	Data Owner	Creator	Modifi
Original records	origin_stroke	[redacted]	[redacted]	Dec 07
Standard records	stand_stroke	[redacted]	[redacted]	Dec 07
Trip facts	stroke_fact	[redacted]	[redacted]	Dec 07
Record statistics	stroke_statistic	[redacted]	[redacted]	Dec 07

Apply for Publication

4 subject selected. 4 subject can be published. Hide

Subject Name	Subject Code	Status
Original records	origin_stroke	● Draft
Standard records	stand_stroke	● Draft
Trip facts	stroke_fact	● Draft
Record statistics	stroke_statistic	● Draft

* Reviewer: --Select--

----End

Creating and Publishing Lookup Tables

This section uses the lookup tables listed in **Table 4-7** as an example.

Table 4-7 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

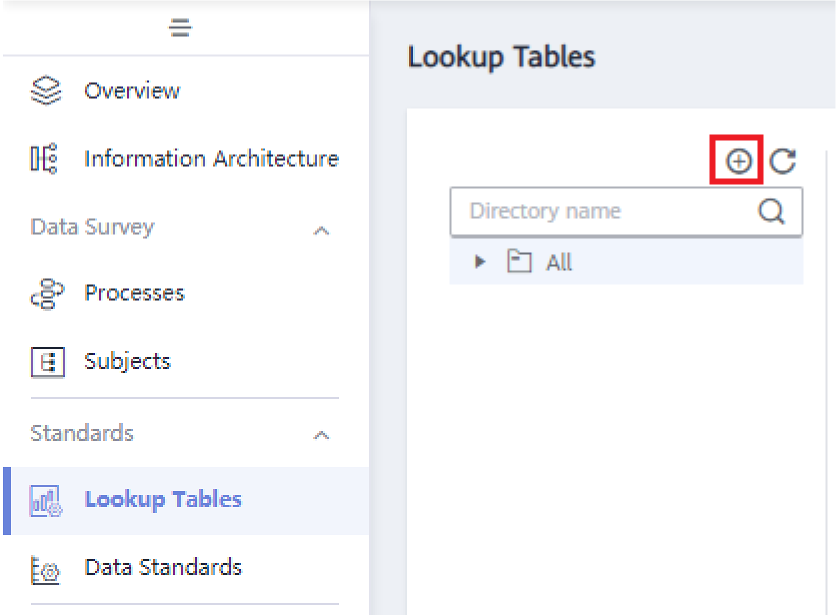
Step 1 On the DataArts Architecture console, choose **Standards** > **Lookup Tables** in the navigation pane on the left.

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.

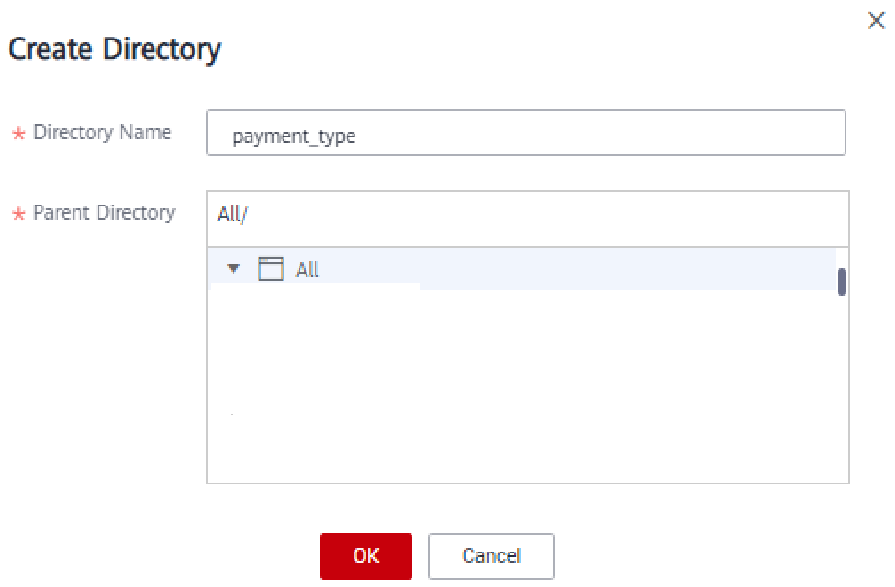
1. On the **Lookup Tables** page, click ⊕ above the directory tree to create a directory.

Figure 4-37 Lookup table directory tree



2. In the dialog box displayed, enter a directory name, select a parent directory, and click **OK**.

Figure 4-38 Creating a directory for lookup tables



* Directory Name: payment_type

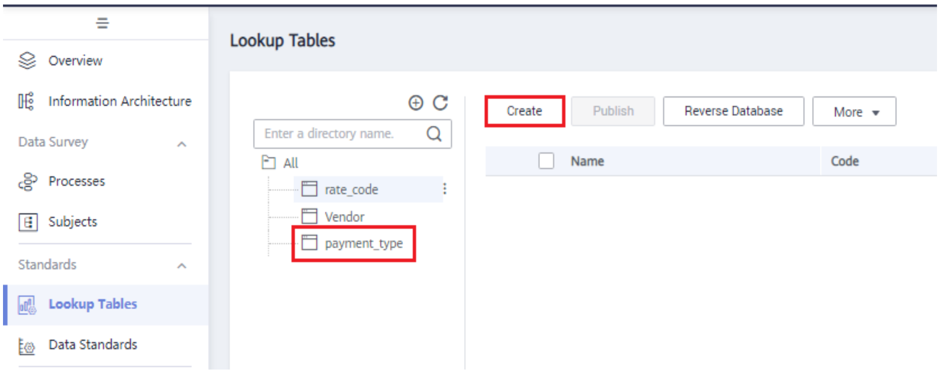
* Parent Directory: All/

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.

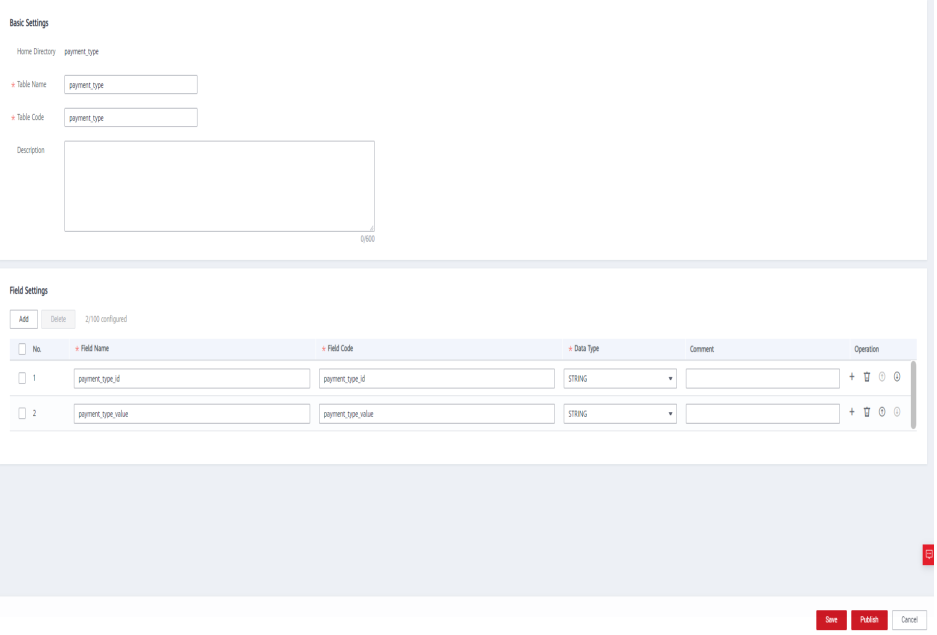
1. On the **Lookup Tables** page, click **payment_type** in the directory tree, and click **Create** on the page displayed.

Figure 4-39 Lookup Tables page



2. Set the parameters based on **Table 4-7** and click **Save**.

Figure 4-40 Creating a lookup table



Basic Settings

Home Directory	payment_type
* Table Name	payment_type
* Table Code	payment_type
Description	

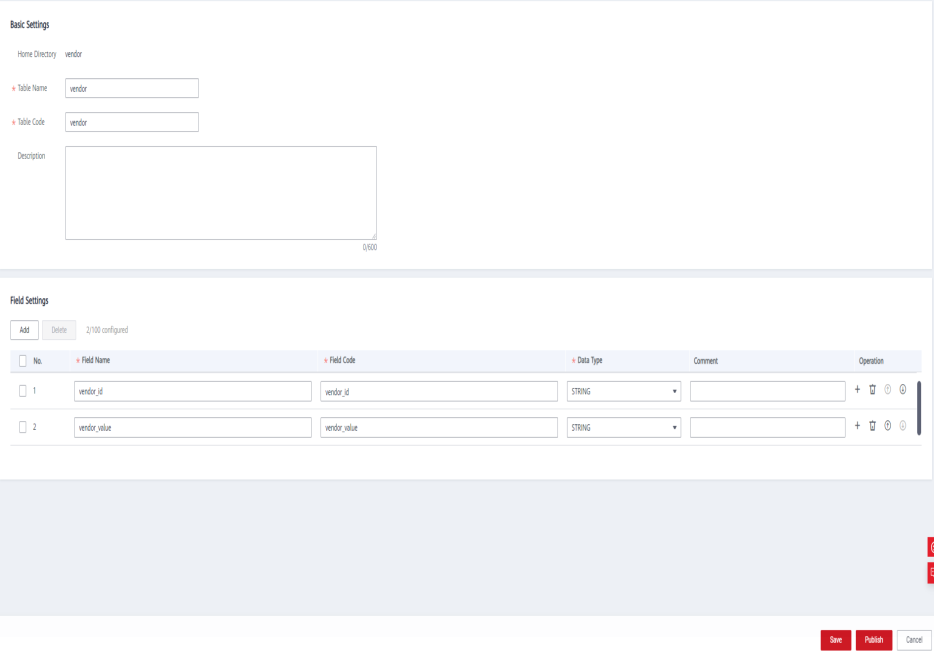
0/600

Field Settings

No.	* Field Name	* Field Code	* Data Type	Comment	Operation
1	payment_type_id	payment_type_id	STRING		
2	payment_type_value	payment_type_value	STRING		

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 4-41 Creating a lookup table named vendor



Basic Settings

Home Directory vendor

* Table Name: vendor

* Table Code: vendor

Description:

0/600

Field Settings

2/100 configured

No.	* Field Name	* Field Code	* Data Type	Comment	Operation
1	vendor_id	vendor_id	STRING		
2	vendor_value	vendor_value	STRING		

Figure 4-42 Creating a lookup table named rate_code

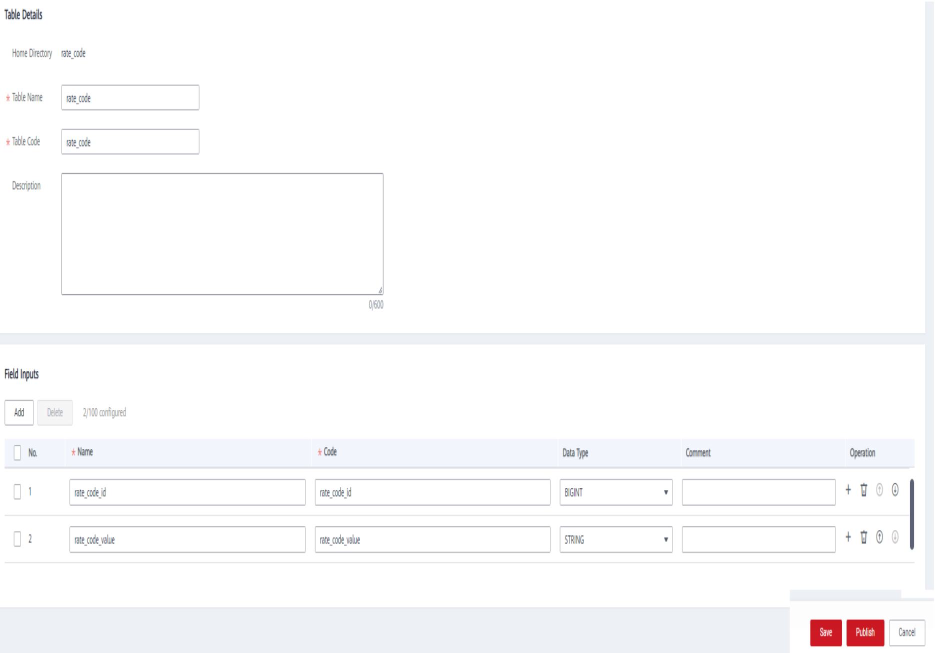


Table Details

Home Directory rate_code

* Table Name: rate_code

* Table Code: rate_code

Description:

0/600

Field Inputs

2/100 configured

No.	* Name	* Code	Data Type	Comment	Operation
1	rate_code_id	rate_code_id	BIGINT		
2	rate_code_value	rate_code_value	STRING		

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 4-8**.

Table 4-8 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 4-9**.

Table 4-9 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 4-10**.

Table 4-10 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 4-11**.

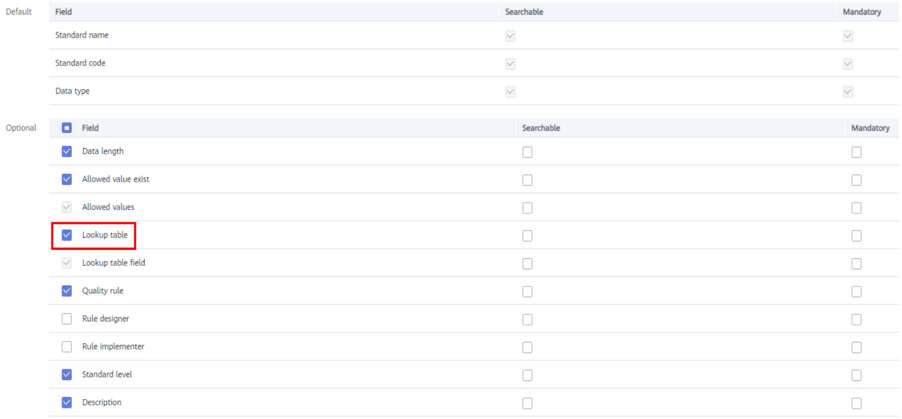
Table 4-11 Data standards

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 4-43 Customize Template



Default

Field	Searchable	Mandatory
Standard name	✓	✓
Standard code	✓	✓
Data type	✓	✓

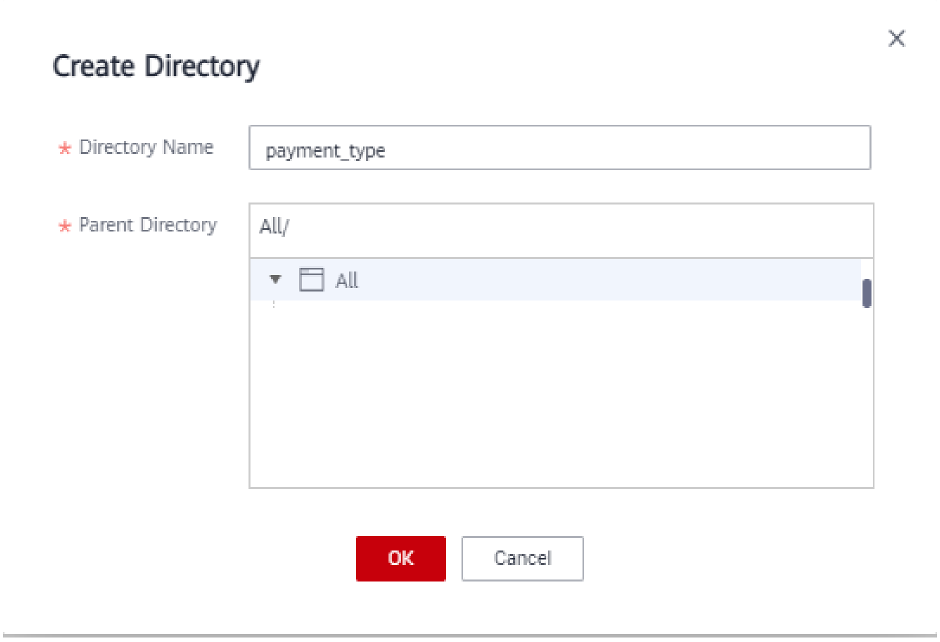
Optional

■ Field	Searchable	Mandatory
☑ Data length	□	□
☑ Allowed value exist	□	□
✓ Allowed values	□	□
☑ Lookup table	□	□
✓ Lookup table field	□	□
☑ Quality rule	□	□
□ Rule designer	□	□
□ Rule implementer	□	□
☑ Standard level	□	□
☑ Description	□	□

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 4-44 Creating a directory for data standards



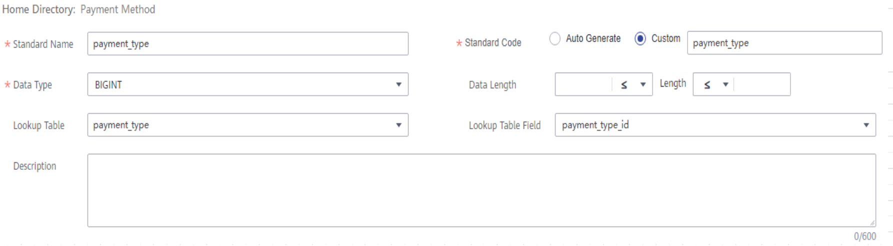
Create Directory

* Directory Name	payment_type
* Parent Directory	All/

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**.

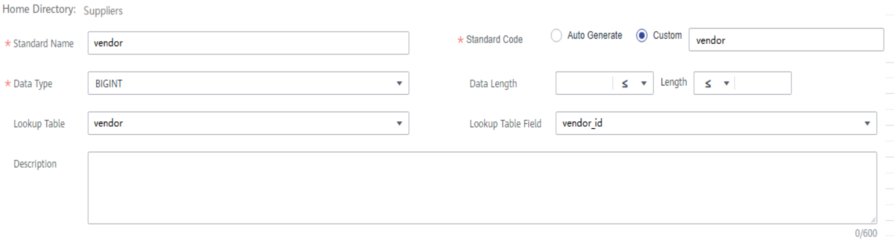
Figure 4-45 Creating a data standard named payment_type



Home Directory: Payment Method

* Standard Name	payment_type	* Standard Code	○ Auto Generate ● Custom payment_type
* Data Type	BIGINT	Data Length	≤ Length ≤
Lookup Table	payment_type	Lookup Table Field	payment_type_id
Description			

Figure 4-46 Creating a data standard named vendor


Home Directory: Suppliers

* Standard Name: vendor

* Standard Code: ○ Auto Generate ● Custom vendor

* Data Type: BIGINT

Data Length: ≤ Length ≤

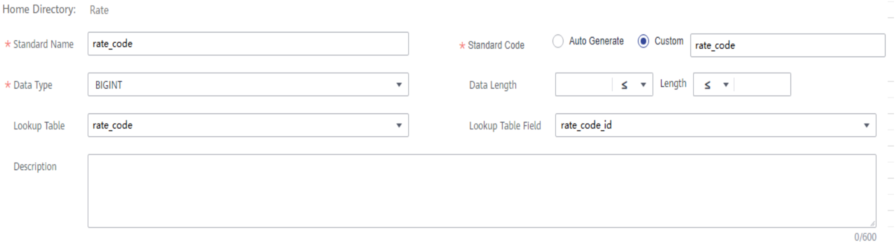
Lookup Table: vendor

Lookup Table Field: vendor_id

Description:

0/600

Figure 4-47 Creating a data standard named rate_code


Home Directory: Rate

* Standard Name: rate_code

* Standard Code: ○ Auto Generate ● Custom rate_code

* Data Type: BIGINT

Data Length: ≤ Length ≤

Lookup Table: rate_code

Lookup Table Field: rate_code_id

Description:

0/600

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

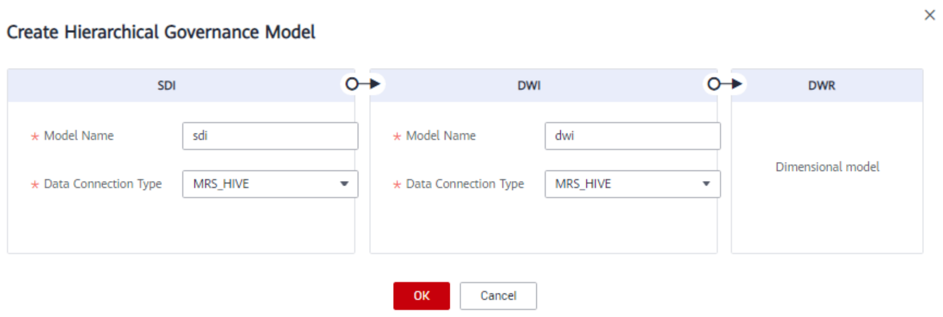
Creating Two ER Models for the SDI and DWI Layers

During ER modeling, create two models for the SDI and DWI layers, import the source table to the ER model for the SDI layer by reversing the database, and create a standard business table to record trip data for the DWI layer.

Step 1 On the DataArts Architecture console, choose **Models** > **ER Modeling** in the navigation pane on the left.

- 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 4-48 Dialog box for creating a hierarchical governance model


Create Hierarchical Governance Model

SDI

* Model Name: sdi

* Data Connection Type: MRS_HIVE

DWI

* Model Name: dwi

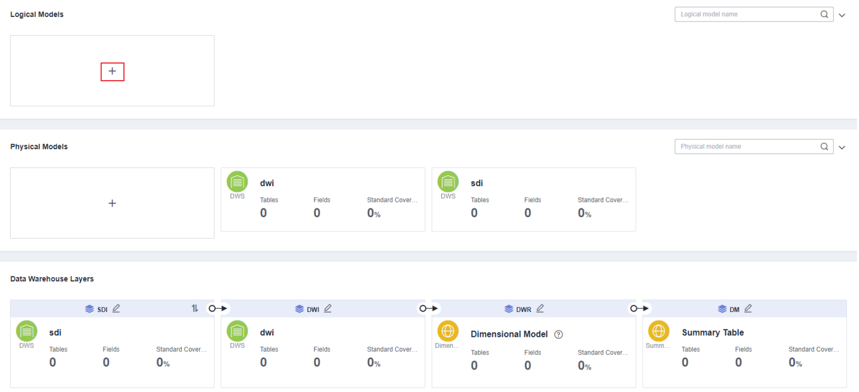
* Data Connection Type: MRS_HIVE

DWR

Dimensional model

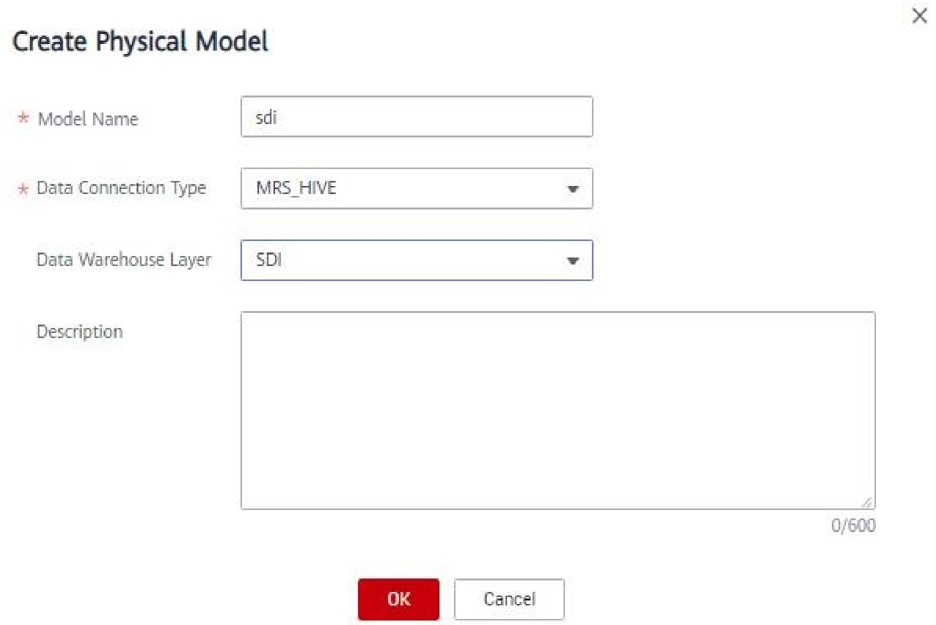
- If you have created ER models before, click + to create physical models, as shown in the following figure.

Figure 4-49 Creating a logical model



a. Create an SDI ER model named **sdi**. Specifically, in the **Physical Models** area, click +, set the parameters as shown in the figure below, and click **OK**.

Figure 4-50 Creating a physical model named sdi



Create Physical Model

* Model Name: sdi

* Data Connection Type: MRS_HIVE

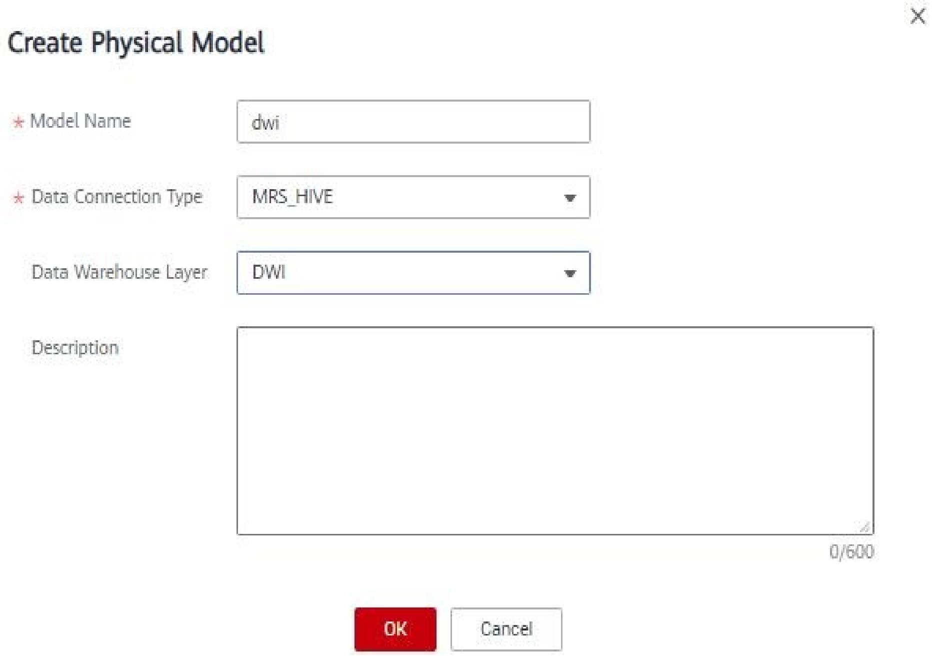
Data Warehouse Layer: SDI

Description:

0/600

b. Create a DWI ER model named **dwi**. Specifically, in the **Physical Models** area, click +, set the parameters as shown in the figure below, and click **OK**.

Figure 4-51 Creating a physical model named dwi



Create Physical Model

* Model Name: dwi

* Data Connection Type: MRS_HIVE

Data Warehouse Layer: DWI

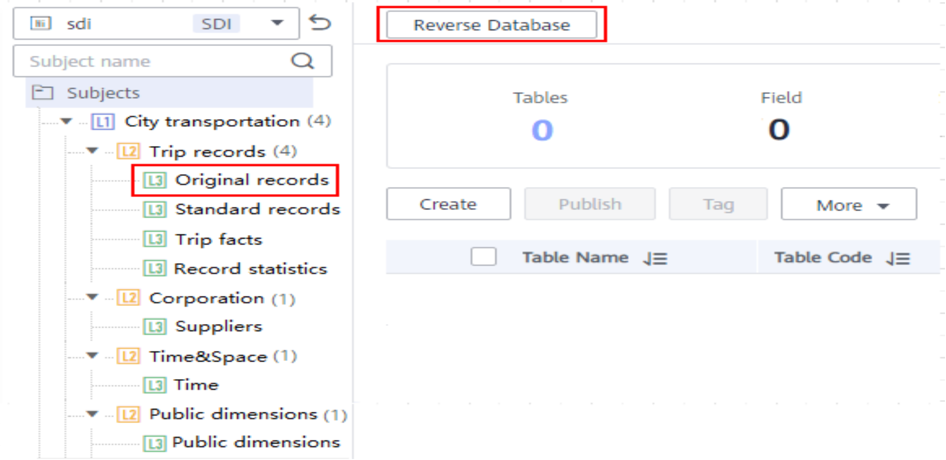
Description:

Step 2 In the **Data Warehouse Layers** area, click the newly created SDI ER model. Choose **City transportation** > **Trip records** > **Original records**, and click **Reverse Database** on the page displayed on the right to import the source table.

NOTE

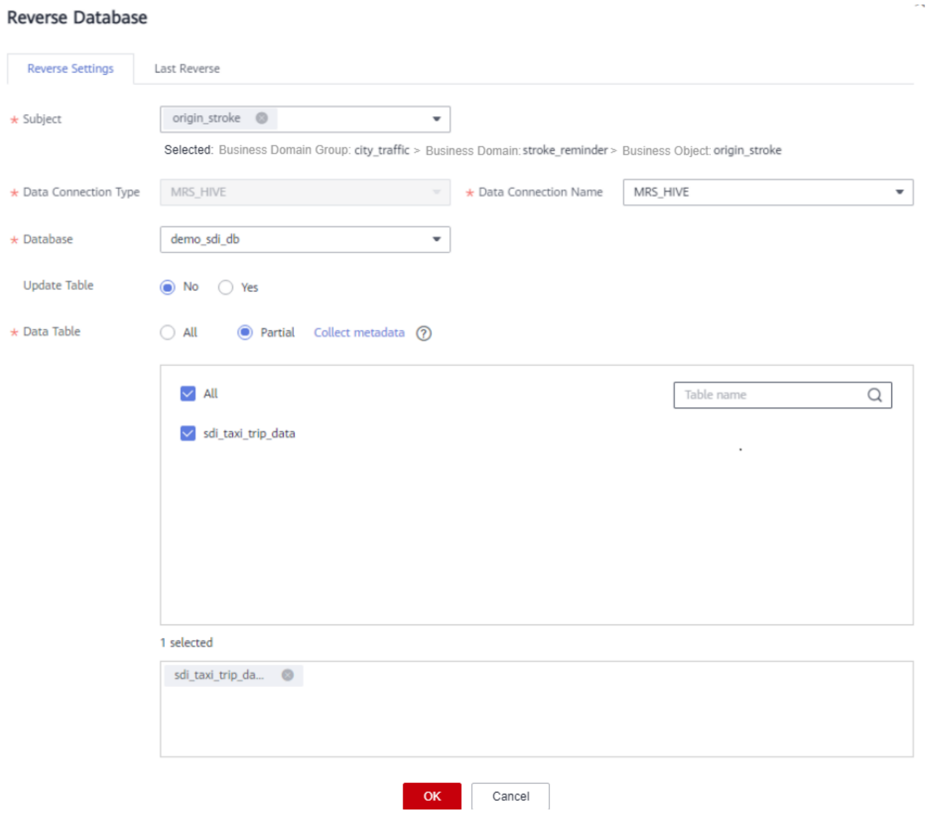
Before reversing a database, ensure that you have collected the data assets of the database. For details on how to collect data assets, see **Step 4: Metadata Collection**.

Figure 4-52 Model directory



In the **Reverse Database** dialog box, set the parameters and click **Yes**. In this example, select the source table in the SDI layer database **demo_sdi_db**.

Figure 4-53 Reversing a database



Reverse Database

* Subject: origin_stroke

Selected: Business Domain Group: city_traffic > Business Domain: stroke_reminder > Business Object: origin_stroke

* Data Connection Type: MRS_HIVE

* Data Connection Name: MRS_HIVE

* Database: demo_sdi_db

Update Table: ● No ○ Yes

* Data Table: ○ All ● Partial Collect metadata

☑ All

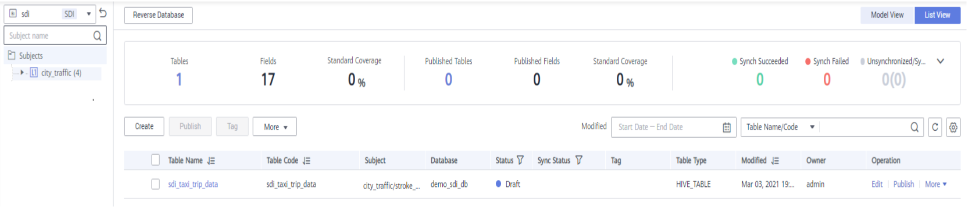
☑ sdi_taxi_trip_data

1 selected

sdi_taxi_trip_da...

After the database is reversed, click **Close**. The table is in the draft state. Click **Publish** in the **Operation** column, and you can view the imported and published table.

Figure 4-54 Viewing a table



Tables	Fields	Standard Coverage	Published Tables	Published Fields	Standard Coverage	Synch Succeeded	Synch Failed	Unsynchronized/Sy...
1	17	0 %	0	0	0 %	0	0	0(0)

	Table Name	Table Code	Subject	Database	Status	Sync Status	Tag	Table Type	Modified	Owner	Operation
□	sdi_taxi_trip_data	sdi_taxi_trip_data	city_traffic/stroke_...	demo_sdi_db	● Draft			HIVE_TABLE	Mar 03, 2021 19:...	admin	Edit | Publish | More

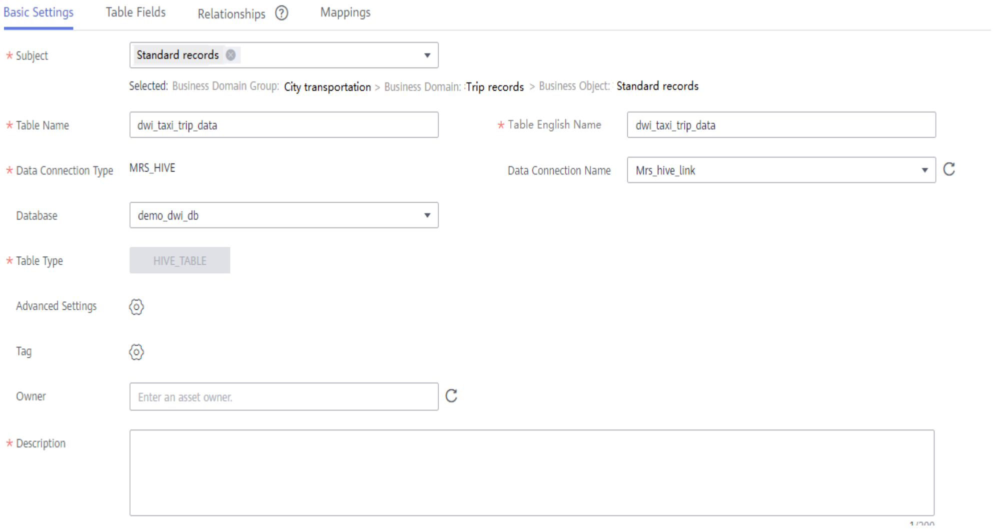
Step 3 Create a standard business table to record trip data.

1. In the **Data Warehouse Layers** area, click the newly created DWI ER model. Choose **City transportation** > **Trip records** > **Standard records**, and click **Create** on the page displayed on the right.
2. On the **Basic Settings** tab page, set the parameters as shown in the figure below.

Table 4-12 Standard trip data table

*Subject	*Table Name	* Table English Name	*Data Connection Name	Database	*Description
Standard records	dwi_taxi_trip_data	dwi_taxi_trip_data	mrs_hive_link	demo_dwi_db	None

Figure 4-55 Basic settings of the table named dwi_taxi_trip_data



* Subject: Standard records
Selected: Business Domain Group: City transportation > Business Domain: Trip records > Business Object: Standard records
* Table Name: dwi_taxi_trip_data
* Table English Name: dwi_taxi_trip_data
* Data Connection Type: MRS_HIVE
Data Connection Name: Mrs_hive_link
Database: demo_dwi_db
* Table Type: HIVE_TABLE
Advanced Settings
Tag
Owner: Enter an asset owner.
* Description

3. Click **Next** to go to the **Table Fields** page. Click **Add**. Add the fields listed in **Table 4-13**. 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 4-56** shows the configuration after the fields are added.

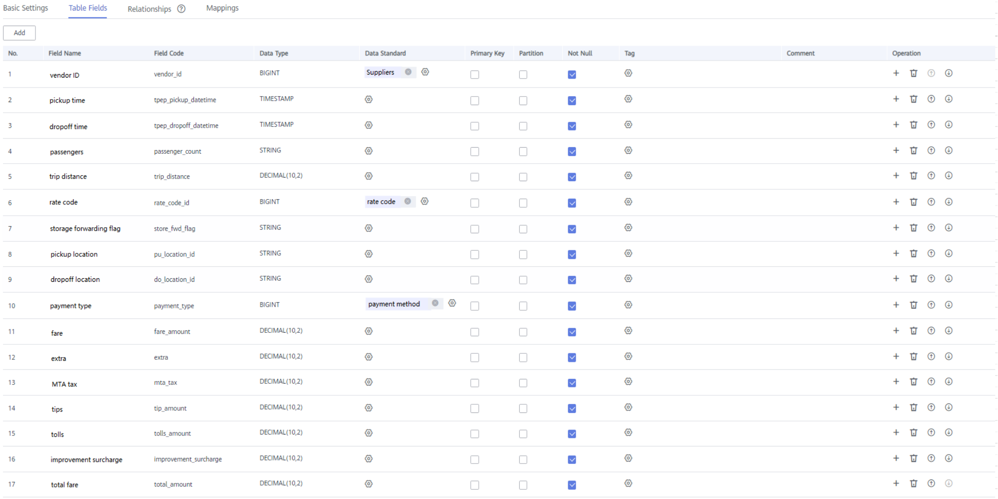
Table 4-13 Fields to be added to the table named dwi_table_trip_data

No.	Field Name	Field Code	Data Type	Data Standard	Primary Key	Partition	Not Null	Tag
1	vendor_id	vendor_id	BIGINT	vendor	Not selected	Not selected	Selected	-
2	tpep_pickup_datetime	tpep_pickup_datetime	TIMESTAMP	-	Not selected	Not selected	Selected	-

No.	Field Name	Field Code	Data Type	Data Standard	Primary Key	Partition	Not Null	Tag
3	tpep_dropoff_datetime	tpep_dropoff_datetime	TIMESTAMP	-	Not selected	Not selected	Selected	-
4	passenger_count	passenger_count	STRING	-	Not selected	Not selected	Selected	-
5	trip_distance	trip_distance	DECIMAL(10,2)	-	Not selected	Not selected	Selected	-
6	rate_code_id	rate_code_id	BIGINT	rate_code	Not selected	Not selected	Selected	-
7	store_fwd_flag	store_fwd_flag	STRING	-	Not selected	Not selected	Selected	-
8	pu_location_id	pu_location_id	STRING	-	Not selected	Not selected	Selected	-
9	do_location_id	do_location_id	STRING	-	Not selected	Not selected	Selected	-
10	payment_type	payment_type	BIGINT	payment_type	Not selected	Not selected	Selected	-
11	fare_amount	fare_amount	DECIMAL(10,2)	-	Not selected	Not selected	Selected	-
12	extra	extra	DECIMAL(10,2)	-	Not selected	Not selected	Selected	-

No.	Field Name	Field Code	Data Type	Data Standard	Primary Key	Partition	Not Null	Tag
13	mta_tax	mta_tax	DECIMAL (10,2)	-	Not selected	Not selected	Selected	-
14	tip_amount	tip_amount	DECIMAL (10,2)	-	Not selected	Not selected	Selected	-
15	tolls_amount	tolls_amount	DECIMAL (10,2)	-	Not selected	Not selected	Selected	-
16	improvement_surcharge	improvement_surcharge	DECIMAL (10,2)	-	Not selected	Not selected	Selected	-
17	total_amount	total_amount	DECIMAL (10,2)	-	Not selected	Not selected	Selected	-

Figure 4-56 Fields to be added to the table named dwi_table_trip_data



No.	Field Name	Field Code	Data Type	Data Standard	Primary Key	Partition	Not Null	Tag	Comment	Operation
1	vendor ID	vendor_id	BIGINT	Suppliers	□	□	☑			
2	pickup time	tpep_pickup_datetime	TIMESTAMP		□	□	☑			
3	dropoff time	tpep_dropoff_datetime	TIMESTAMP		□	□	☑			
4	passengers	passenger_count	STRING		□	□	☑			
5	trip distance	trip_distance	DECIMAL(10,2)		□	□	☑			
6	rate code	rate_code_id	BIGINT	rate code	□	□	☑			
7	storage forwarding flag	store_fwd_flag	STRING		□	□	☑			
8	pickup location	pu_location_id	STRING		□	□	☑			
9	dropoff location	do_location_id	STRING		□	□	☑			
10	payment type	payment_type	BIGINT	payment method	□	□	☑			
11	fare	fare_amount	DECIMAL(10,2)		□	□	☑			
12	extra	extra	DECIMAL(10,2)		□	□	☑			
13	MTA tax	mta_tax	DECIMAL(10,2)		□	□	☑			
14	tips	tip_amount	DECIMAL(10,2)		□	□	☑			
15	tolls	tolls_amount	DECIMAL(10,2)		□	□	☑			
16	improvement surcharge	improvement_surcharge	DECIMAL(10,2)		□	□	☑			
17	total fare	total_amount	DECIMAL(10,2)		□	□	☑			

You can perform the following operations on the fields.

- **Associating with data standards**

  When creating or editing a table, click the **Table Fields** tab. In the **Data Standard** column of the row where the field is located, click ⚙ to select a data standard to be associated with the field. After a field is associated with a data standard, a quality job is automatically generated after the table is published. A quality rule is generated for each field associated with the data standard. You can monitor the quality of fields based on the data standard. You can view the quality job on the **Quality Jobs** page of **DataArts Quality**. For more information about associating data standards, see **Designing Physical Models**.

- **Adding a tag**

  A tag is a custom identifier. After adding a tag, you can search for data assets in the DataArts Studio DataArts Catalog module with ease.

  When creating or editing a table, click the **Table Fields** tab. In the **Tag** column of the row where the field is located, click ⚙ to select a tag. In the dialog box displayed, enter a new tag name and press **Enter**. Alternatively, select an existing tag from the drop-down list. Then click **OK**.

- **Associating with quality rules**

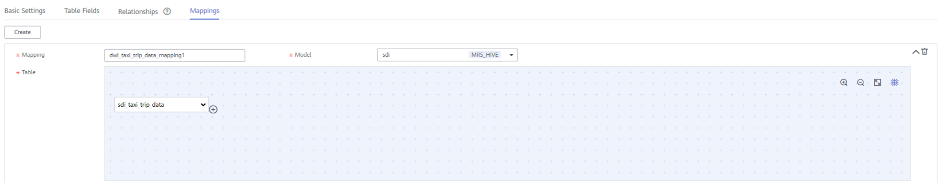
  After a table is created, you can associate fields in the table with quality rules. After the association, a quality job is automatically created in the DataArts Quality module after the table is published. If the table has been published, the system automatically updates the quality job. For more information about associating quality rules, see **Associating with Quality Rules**.

4. Click **Next** to go to the **Relationships** page. In this example, you do not need to perform any operation on this page.
5. Click **Next** to go to the **Mappings** page and create mappings to design data sources of the table.
   - If the table fields come from different ER 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. In the **Table** field of the mapping, you can join multiple tables and then set source fields for the fields in the table.

   In this example, you only need to create one mapping. Click **Create** and set a mapping as shown in **Figure 4-57**.

   - **Mapping** is automatically generated. You can customize the name.
   - Select **sdi** for **Model**.
   - Select the source table **sdi_taxi_trip_data** for **Table**. All data in the **dwi_taxi_trip_data** table comes from this source table.

Figure 4-57 Creating a mapping



* Mapping: dwi_taxi_trip_data_mapping1
* Model: sdi MRS_HIVE
* Table: sdi_taxi_trip_data

   - **Field Mapping**

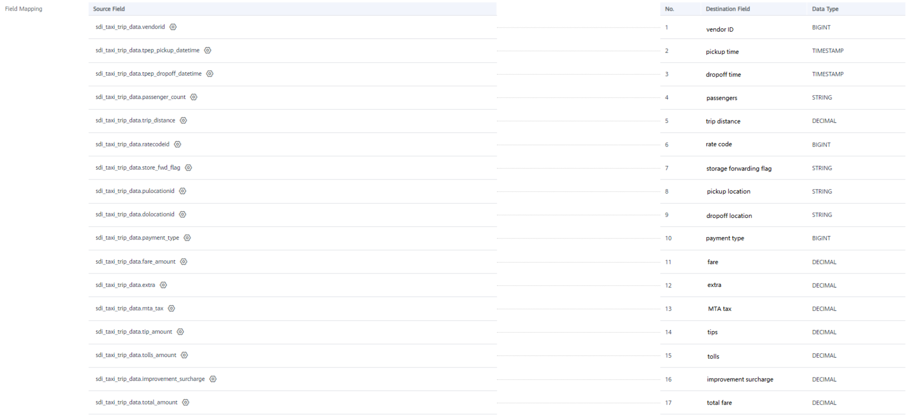
In the **Field Mapping** area, set source fields for the fields in the table in sequence. The selected source fields must have the same meaning as the fields in the table. As shown in **Figure 4-58**, an SQL statement is displayed at the bottom of **Field Mapping** for reference.

NOTE

- On the **DataArts Architecture** page, choose **Metrics** > **Configuration Center** in the navigation pane on the left, and click the **Functions** tab. On the **Functions** page, if **Create data development jobs** is selected (unselected by default) for **Model Design Process**, the system can create 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 4-58 Mapping fields



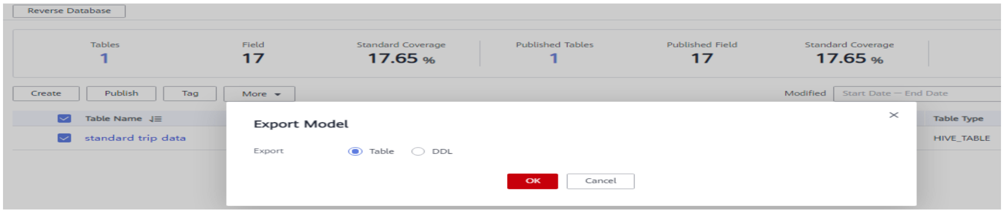
Field Mapping

Source Field	No.	Destination Field	Data Type
sdi_taxi_trip_data.vendorid	1	vendor ID	BIGINT
sdi_taxi_trip_data.tpep_pickup_datetime	2	pickup time	TIMESTAMP
sdi_taxi_trip_data.tpep_dropoff_datetime	3	dropoff time	TIMESTAMP
sdi_taxi_trip_data.passenger_count	4	passengers	STRING
sdi_taxi_trip_data.trip_distance	5	trip distance	DECIMAL
sdi_taxi_trip_data.ratecodeid	6	rate code	BIGINT
sdi_taxi_trip_data.store_fwd_flag	7	storage forwarding flag	STRING
sdi_taxi_trip_data.pulocationid	8	pickup location	STRING
sdi_taxi_trip_data.dolocationid	9	dropoff location	STRING
sdi_taxi_trip_data.payment_type	10	payment type	BIGINT
sdi_taxi_trip_data.fare_amount	11	fare	DECIMAL
sdi_taxi_trip_data.extra	12	extra	DECIMAL
sdi_taxi_trip_data.mta_tax	13	MTA tax	DECIMAL
sdi_taxi_trip_data.tip_amount	14	tips	DECIMAL
sdi_taxi_trip_data.tolls_amount	15	tolls	DECIMAL
sdi_taxi_trip_data.improvement_surcharge	16	improvement surcharge	DECIMAL
sdi_taxi_trip_data.total_amount	17	total fare	DECIMAL

6. After the mappings are configured, click **Save**.

Step 4 Select the created model and choose **More** > **Export**. In the dialog box displayed, select **Table** for **Export** and click **OK**. Export the **sdi** model in the same way. You can use the exported model as a backup and import it.

Figure 4-59 Export dialog box



Tables	Field	Standard Coverage	Published Tables	Published Field	Standard Coverage
1	17	17.65 %	1	17	17.65 %

☑	Table Name	Table Type
☑	standard trip data	HIVE_TABLE

Export Model

Export ● Table ○ DDL

Step 5 Publish the table model.

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** of the corresponding model on the **ER Modeling** page.

Publication is an asynchronous operation. You can click ⟳ to refresh the status. After an application for publishing a table is approved, the system performs operations such as creating tables and synchronizing technical assets and logical assets based on the configurations of **Model Design Process** on the **Functions** tab page in **Configuration Center**. The synchronization status is displayed in the **Sync Status** column of the table.

- If all items in **Sync Status** are displayed as **Succeeded**, the table is published.

  Move your mouse pointer to ▦ in **Sync Status**. If **Creation succeeded** is displayed, the table is created in the corresponding data source.

- If an item in **Sync Status** is displayed as **Failed**, you can refresh the status. If the fault persists, choose **More** > **View History** to view logs.

  Locate the failure cause based on the logs. After the fault is rectified, return to the **ER Modeling** page, select the table to be synchronized in the list, choose **More** > **Synchronize** and click **OK** in the dialog box displayed. If the synchronization fails again, contact technical support for assistance.

Figure 4-60 Checking the table status

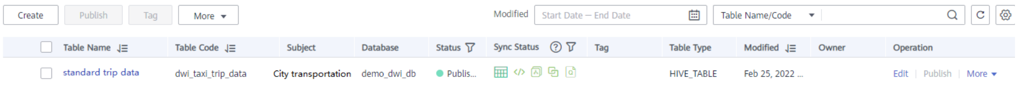
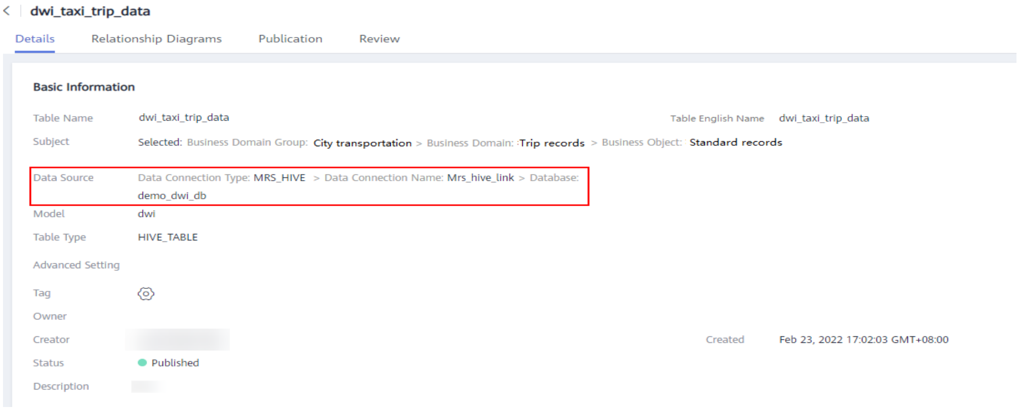


	Table Name	Table Code	Subject	Database	Status	Sync Status	Tag	Table Type	Modified	Owner	Operation
□	standard trip data	dwi_taxi_trip_data	City transportation	demo_dwi_db	● Publis...	▦		HIVE_TABLE	Feb 25, 2022 ...		Edit | Publish | More

Click a table name in the list to view the table details. **Data Source** shows the table location.

Figure 4-61 Table details



dwi_taxi_trip_data

Basic Information

Table Name dwi_taxi_trip_data

Table English Name dwi_taxi_trip_data

Subject Selected: Business Domain Group: City transportation > Business Domain: :Trip records > Business Object: Standard records

Data Source Data Connection Type: MRS_HIVE > Data Connection Name: Mrs_hive_link > Database: demo_dwi_db

Model dwi

Table Type HIVE_TABLE

Advanced Setting

Tag

Owner

Creator

Created Feb 23, 2022 17:02:03 GMT+08:00

Status ● Published

Description

----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 4-14**.

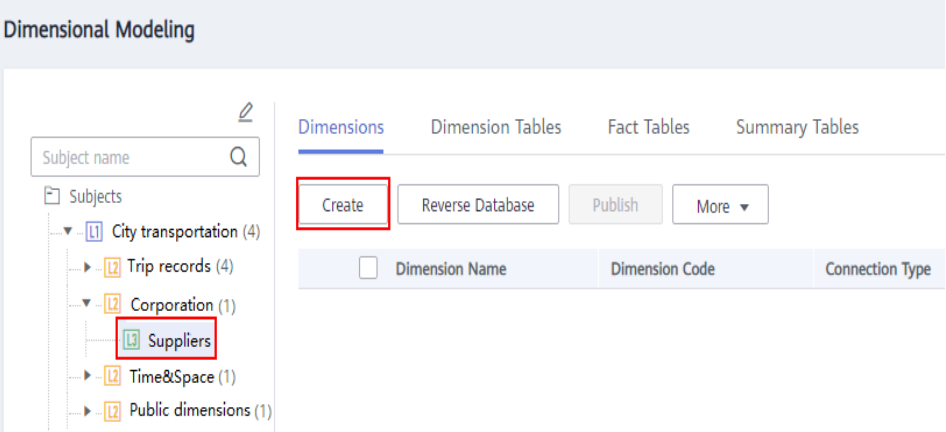
Table 4-14 Lookup table dimensions

*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

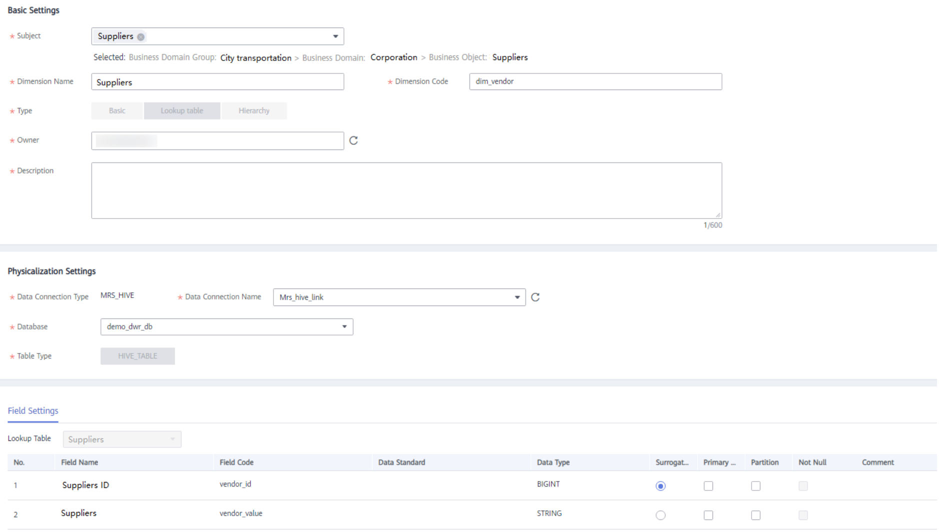
1. Click the **Dimensions** tab, choose **City transportation** > **Corporation** > **Suppliers** in the subject tree, and click **Create** to create a dimension named **dim_vendor**.

Figure 4-62 Dimensional modeling



2. On the **Create Dimension** page, set the parameters as shown in the figure below and click **Save**.

Figure 4-63 Creating a dimension named dim_vendor



Basic Settings

* Subject: Suppliers

Selected: Business Domain Group: City transportation > Business Domain: Corporation > Business Object: Suppliers

* Dimension Name: Suppliers

* Dimension Code: dim_vendor

* Type: Basic | Lookup table | Hierarchy

* Owner: [redacted]

* Description:

1/600

Physicalization Settings

* Data Connection Type: MRS_HIVE

* Data Connection Name: Mrs_hive_link

* Database: demo_dwr_db

* Table Type: HIVE_TABLE

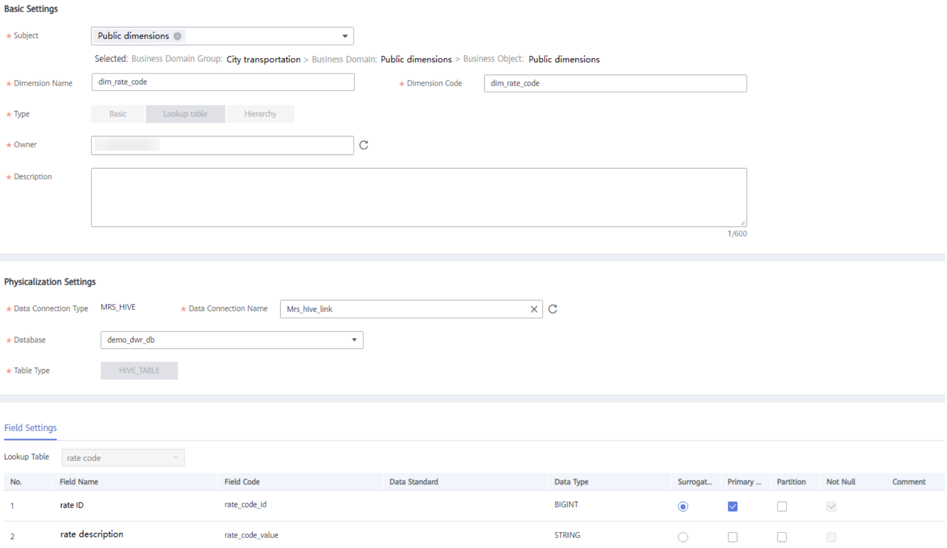
Field Settings

Lookup Table: Suppliers

No.	Field Name	Field Code	Data Standard	Data Type	Surrogat...	Primary ...	Partition	Not Null	Comment
1	Suppliers ID	vendor_id		BIGINT	●	□	□	□	
2	Suppliers	vendor_value		STRING	○	□	□	□	

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 4-64 Creating a dimension named dim_rate_code



Basic Settings

* Subject: Public dimensions

Selected: Business Domain Group: City transportation > Business Domain: Public dimensions > Business Object: Public dimensions

* Dimension Name: dim_rate_code

* Dimension Code: dim_rate_code

* Type: Basic | Lookup table | Hierarchy

* Owner: [redacted]

* Description:

1/600

Physicalization Settings

* Data Connection Type: MRS_HIVE

* Data Connection Name: Mrs_hive_link

* Database: demo_dwr_db

* Table Type: HIVE_TABLE

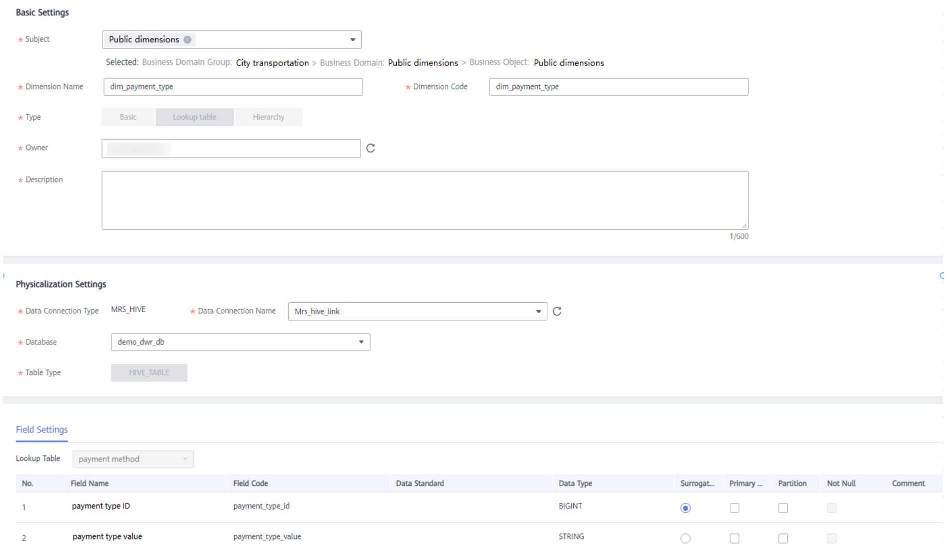
Field Settings

Lookup Table: rate code

No.	Field Name	Field Code	Data Standard	Data Type	Surrogat...	Primary ...	Partition	Not Null	Comment
1	rate ID	rate_code_id		BIGINT	●	☑	□	☑	
2	rate description	rate_code_value		STRING	○	□	□	□	

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 4-65 Creating a dimension named dim_payment_type



Basic Settings

* Subject: Public dimensions

Selected: Business Domain Group: City transportation > Business Domain: Public dimensions > Business Object: Public dimensions

* Dimension Name: dim_payment_type   * Dimension Code: dim_payment_type

* Type: Basic | Lookup table | Hierarchy

* Owner: [redacted]

* Description:

1/600

Physicalization Settings

* Data Connection Type: MRS_HIVE   * Data Connection Name: Mrs_hive_link

* Database: demo_dwr_db

* Table Type: HIVE_TABLE

Field Settings

Lookup Table: payment method

No.	Field Name	Field Code	Data Standard	Data Type	Surrogat...	Primary ...	Partition	Not Null	Comment
1	payment type ID	payment_type_id		BIGINT	●	□	□	□	
2	payment type value	payment_type_value		STRING	○	□	□	□	

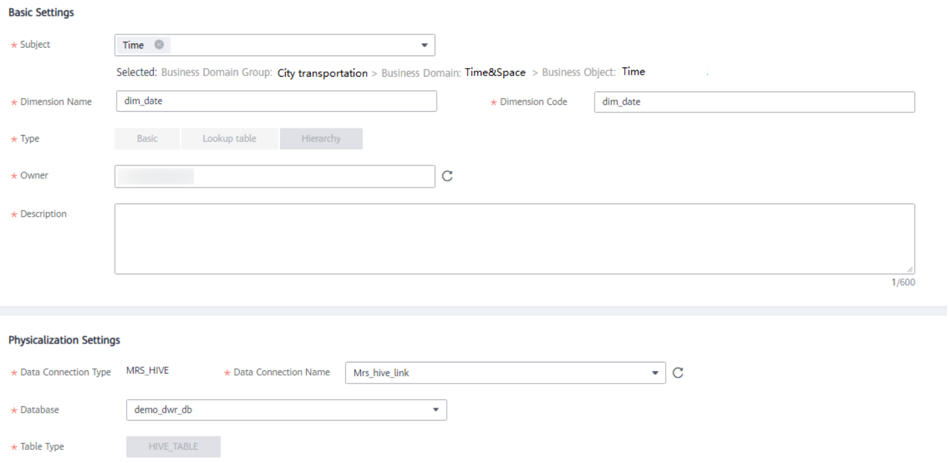
Step 3 Create a hierarchy dimension named **dim_date**.

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 4-15 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 4-66 Date dimension



Basic Settings

* Subject: Time

Selected: Business Domain Group: City transportation > Business Domain: Time&Space > Business Object: Time

* Dimension Name: dim_date   * Dimension Code: dim_date

* Type: Basic | Lookup table | Hierarchy

* Owner: [redacted]

* Description:

1/600

Physicalization Settings

* Data Connection Type: MRS_HIVE   * Data Connection Name: Mrs_hive_link

* Database: demo_dwr_db

* Table Type: HIVE_TABLE

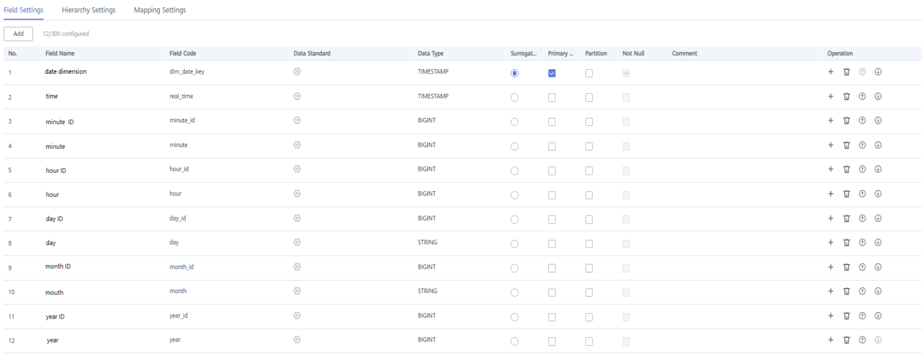
3. In the **Field Settings** area, add fields as described in the table below.

Table 4-16 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 4-67 Field settings



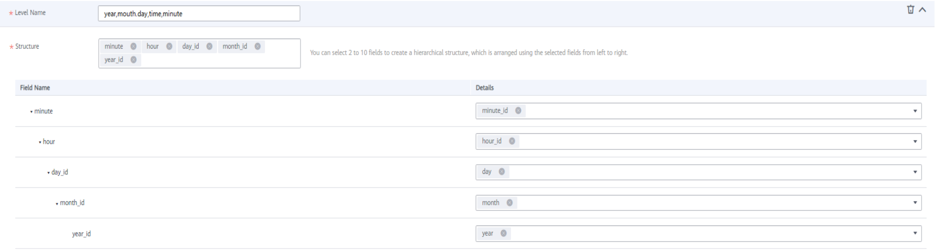
Field Settings | Hierarchy Settings | Mapping Settings

Add 12/300 configured

No.	Field Name	Field Code	Data Standard	Data Type	Surrogat...	Primary ...	Partition	Not Null	Comment	Operation
1	date dimension	dim_date_key	⚙	TIMESTAMP	●	☑	□	✓		+ 🗑 ⊕ ⊕
2	time	real_time	⚙	TIMESTAMP	○	□	□	□		+ 🗑 ⊕ ⊕
3	minute ID	minute_id	⚙	BIGINT	○	□	□	□		+ 🗑 ⊕ ⊕
4	minute	minute	⚙	BIGINT	○	□	□	□		+ 🗑 ⊕ ⊕
5	hour ID	hour_id	⚙	BIGINT	○	□	□	□		+ 🗑 ⊕ ⊕
6	hour	hour	⚙	BIGINT	○	□	□	□		+ 🗑 ⊕ ⊕
7	day ID	day_id	⚙	BIGINT	○	□	□	□		+ 🗑 ⊕ ⊕
8	day	day	⚙	STRING	○	□	□	□		+ 🗑 ⊕ ⊕
9	month ID	month_id	⚙	BIGINT	○	□	□	□		+ 🗑 ⊕ ⊕
10	mouth	month	⚙	STRING	○	□	□	□		+ 🗑 ⊕ ⊕
11	year ID	year_id	⚙	BIGINT	○	□	□	□		+ 🗑 ⊕ ⊕
12	year	year	⚙	BIGINT	○	□	□	□		+ 🗑 ⊕ ⊕

4. In the **Hierarchy Settings** area, click **Add** to create two layers as shown in the figures below.

Figure 4-68 Layer 1



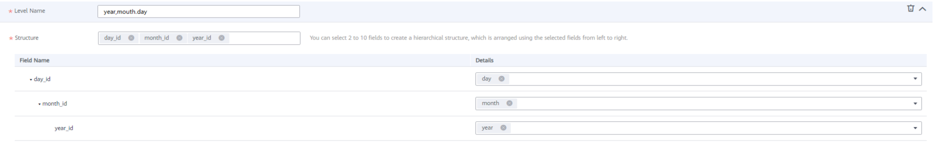
* Level Name: year,mouth.day,time,minute

* Structure: minute, hour, day_id, month_id, year_id

You can select 2 to 10 fields to create a hierarchical structure, which is arranged using the selected fields from left to right.

Field Name	Details
minute	minute_id
hour	hour_id
day_id	day
month_id	month
year_id	year

Figure 4-69 Layer 2



* Level Name: year,mouth.day

* Structure: day_id, month_id, year_id

You can select 2 to 10 fields to create a hierarchical structure, which is arranged using the selected fields from left to right.

Field Name	Details
day_id	day
month_id	month
year_id	year

5. Click **Save**.

Step 4 Return to the **Dimensions** tab page, select the four new dimensions in the dimension list, and click **Publish**.

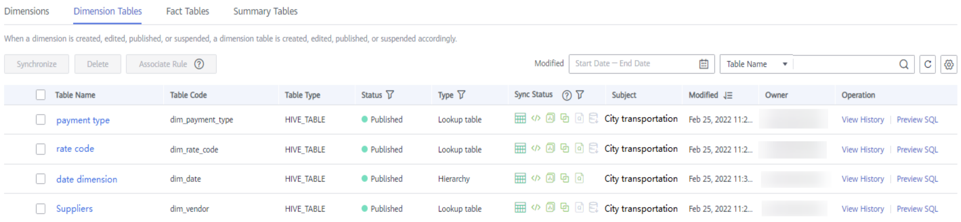
Step 5 In the **Apply for Publication** dialog box, select a reviewer and click **OK**. Wait for the reviewer to review the application. If you have the reviewer permissions, select **Auto-review** and click **OK**.

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 4-70 Sync Status of the dimension tables



When a dimension is created, edited, published, or suspended, a dimension table is created, edited, published, or suspended accordingly.

Table Name	Table Code	Table Type	Status	Type	Sync Status	Subject	Modified	Owner	Operation
payment type	dim_payment_type	HIVE_TABLE	Published	Lookup table		City transportation	Feb 25, 2022 11:2...	[redacted]	View History | Preview SQL
rate code	dim_rate_code	HIVE_TABLE	Published	Lookup table		City transportation	Feb 25, 2022 11:2...	[redacted]	View History | Preview SQL
date dimension	dim_date	HIVE_TABLE	Published	Hierarchy		City transportation	Feb 25, 2022 11:3...	[redacted]	View History | Preview SQL
Suppliers	dim_vendor	HIVE_TABLE	Published	Lookup table		City transportation	Feb 25, 2022 11:2...	[redacted]	View History | Preview SQL

----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 4-17**.

Table 4-17 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

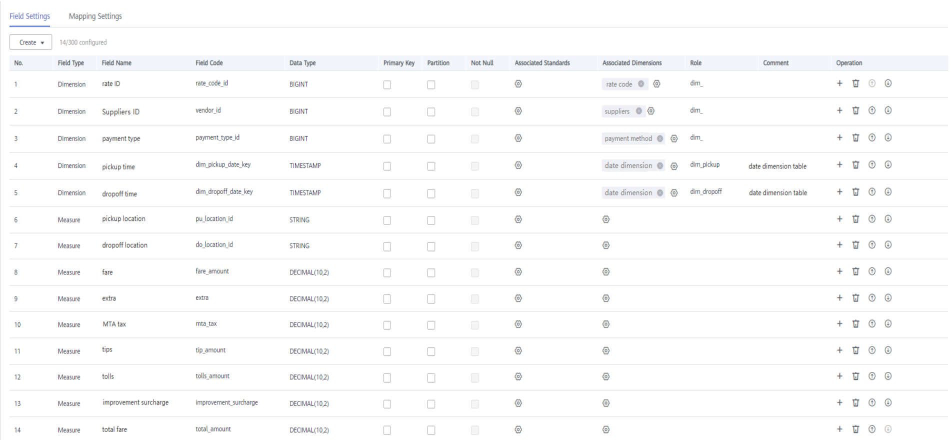
In the **Field Settings** area, choose **Create** > **Measure** and create the fields listed in **Table 4-18** in sequence.

Table 4-18 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 4-71 Fact table fields



No.	Field Type	Field Name	Field Code	Data Type	Primary Key	Partition	Not Null	Associated Standards	Associated Dimensions	Role	Comment	Operation
1	Dimension	rate ID	rate_code_id	BIGINT	□	□	□		rate code	dim_		
2	Dimension	Suppliers ID	vendor_id	BIGINT	□	□	□		suppliers	dim_		
3	Dimension	payment type	payment_type_id	BIGINT	□	□	□		payment method	dim_		
4	Dimension	pickup time	dim_pickup_date_key	TIMESTAMP	□	□	□		date dimension	dim_pickup	date dimension table	
5	Dimension	dropoff time	dim_dropoff_date_key	TIMESTAMP	□	□	□		date dimension	dim_dropoff	date dimension table	
6	Measure	pickup location	pu_location_id	STRING	□	□	□					
7	Measure	dropoff location	do_location_id	STRING	□	□	□					
8	Measure	fare	fare_amount	DECIMAL(10,2)	□	□	□					
9	Measure	extra	extra	DECIMAL(10,2)	□	□	□					
10	Measure	MTA tax	mta_tax	DECIMAL(10,2)	□	□	□					
11	Measure	tips	tip_amount	DECIMAL(10,2)	□	□	□					
12	Measure	tolls	tolls_amount	DECIMAL(10,2)	□	□	□					
13	Measure	improvement surcharge	improvement_surcharge	DECIMAL(10,2)	□	□	□					
14	Measure	total fare	total_amount	DECIMAL(10,2)	□	□	□					

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 4-19** and **Table 4-20**.

Table 4-19 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 4-20 Derivative metrics

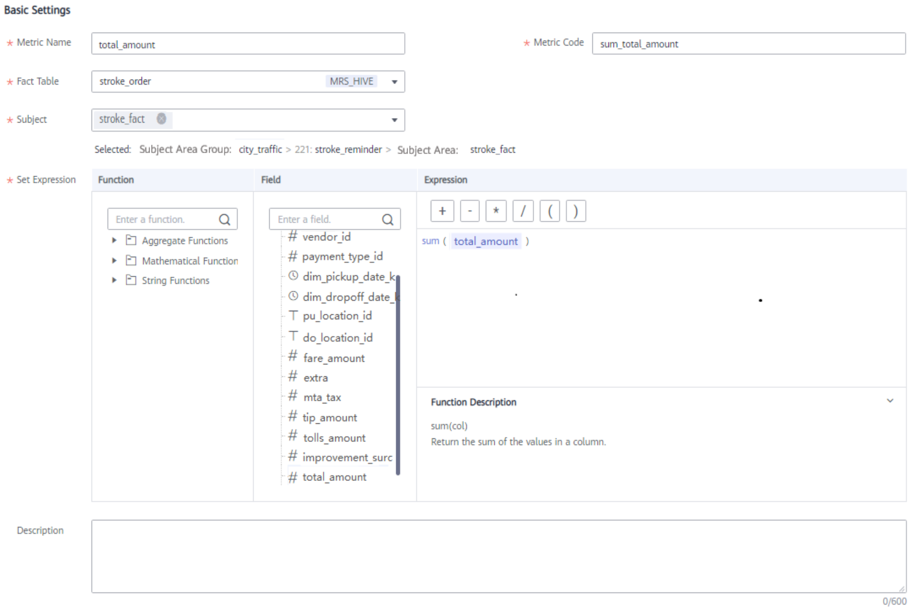
Metric	*Data Table	*Subject	*Atomic Metric	Statistical Dimension	Time Filter	General Filter
total_amount_(payment_type)	Itinerary order	stroke_statistic	total_amount	payment_type	None	None
total_amount_(rate_code)	Itinerary order	stroke_statistic	total_amount	rate_code	None	None
total_amount_(vendor,stroke_order.dim_dropoff_date_key)	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 4-72 Creating an atomic metric



Basic Settings

* Metric Name: total_amount

* Metric Code: sum_total_amount

* Fact Table: stroke_order MRS_HIVE

* Subject: stroke_fact

Selected: Subject Area Group: city_traffic > 221: stroke_reminder > Subject Area: stroke_fact

* Set Expression

Expression: sum (total_amount)

Function Description

sum(col)

Return the sum of the values in a column.

Description: 0/600

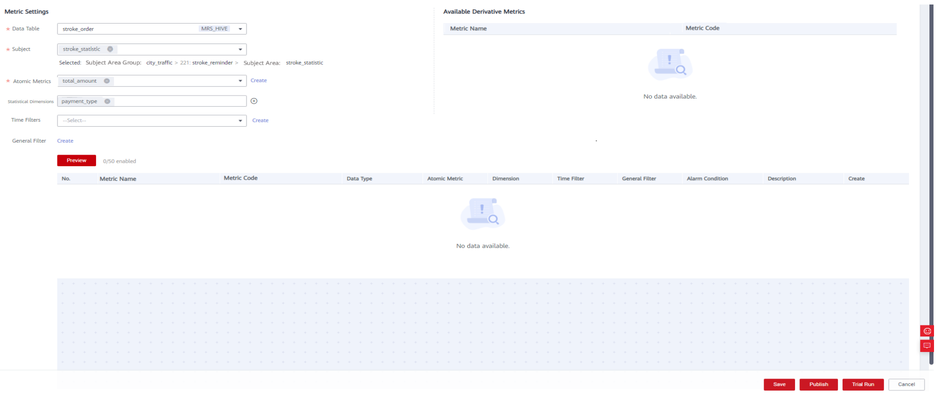
3. Wait for the reviewer to review the application. After the application is approved, the atomic metric will be created.

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 4-73 Creating a derivative metric named total_amount_(payment_type)



Metric Settings

* Data Table: stroke_order MRS_HIVE

* Subject: stroke_statistic

Selected: Subject Area Group: city_traffic > 221: stroke_reminder > Subject Area: stroke_statistic

* Atomic Metrics: total_amount

Statistical Dimensions: payment_type

Time Filters: --Select--

General Filter

0/50 enabled

No.	Metric Name	Metric Code	Data Type	Atomic Metric	Dimension	Time Filter	General Filter	Alarm Condition	Description	Create

No data available.

Available Derivative Metrics

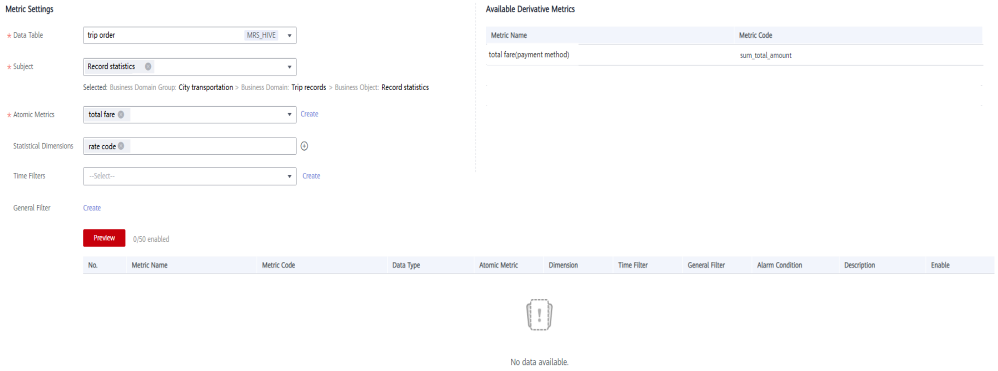
Metric Name	Metric Code

No data available.

- 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 4-74 Creating a derivative metric named total_amount_(rate_code)

  Metric Settings

  * Data Table: trip order MRS_HIVE

  * Subject: Record statistics

  Selected: Business Domain Group: City transportation > Business Domain: Trip records > Business Object: Record statistics

  * Atomic Metrics: total fare Create

  Statistical Dimensions: rate code

  Time Filters: --Select-- Create

  General Filter: Create

  Preview 0/50 enabled

No.	Metric Name	Metric Code	Data Type	Atomic Metric	Dimension	Time Filter	General Filter	Alarm Condition	Description	Enable

  No data available.

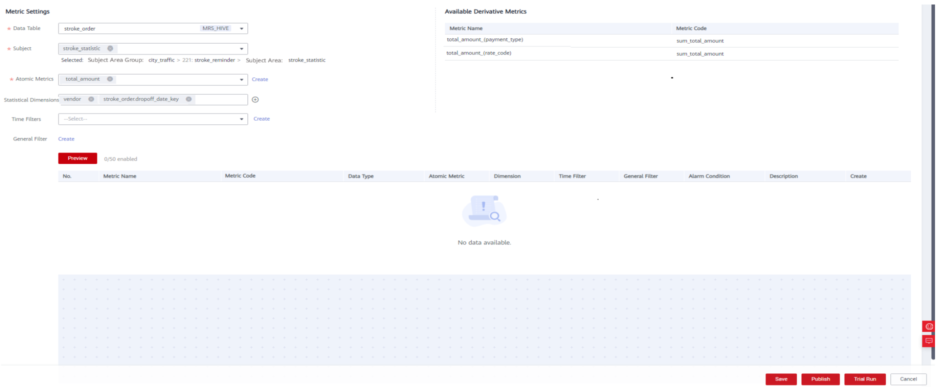
  Available Derivative Metrics

Metric Name	Metric Code
total fare(payment method)	sum_total_amount

- 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 4-75 Creating a derivative metric named total_amount_(vendor,stroke_order.dim_dropoff_date_key)

  Metric Settings

  * Data Table: stroke_order MRS_HIVE

  * Subject: stroke_statistic

  Selected: Subject Area Group: city_traffic > 221: stroke_reminder > Subject Area: stroke_statistic

  * Atomic Metrics: total_amount Create

  Statistical Dimensions: vendor, stroke_order.dropoff_date_key

  Time Filters: --Select-- Create

  General Filter: Create

  Preview 0/50 enabled

No.	Metric Name	Metric Code	Data Type	Atomic Metric	Dimension	Time Filter	General Filter	Alarm Condition	Description	Create

  No data available.

  Available Derivative Metrics

Metric Name	Metric Code
total_amount_(payment_type)	sum_total_amount
total_amount_(rate_code)	sum_total_amount

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 4-21** for the DM layer.

Table 4-21 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.

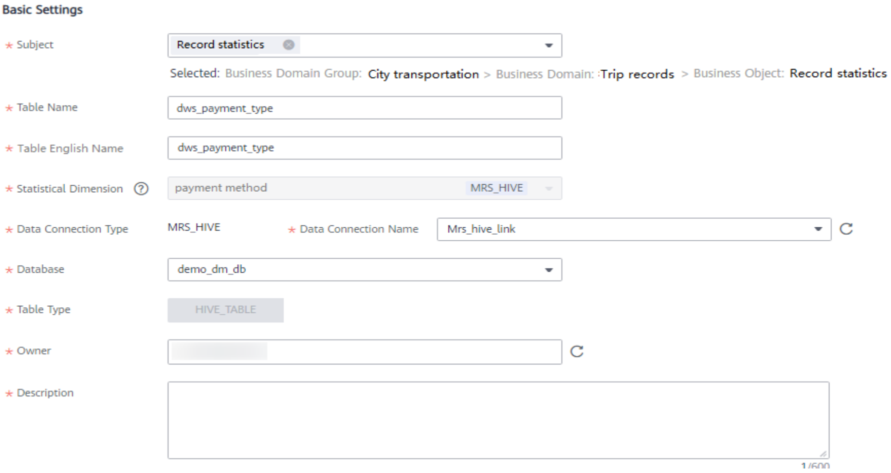
Step 2 Click the **Summary Tables** tab.

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 4-76 Creating a summary table named dws_payment_type



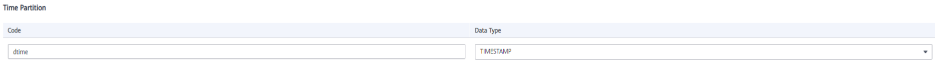
Basic Settings

* Subject	Record statistics
	Selected: Business Domain Group: City transportation > Business Domain: Trip records > Business Object: Record statistics
* Table Name	dws_payment_type
* Table English Name	dws_payment_type
* Statistical Dimension	payment method MRS_HIVE
* Data Connection Type	MRS_HIVE
* Data Connection Name	Mrs_hive_link
* Database	demo_dm_db
* Table Type	HIVE_TABLE
* Owner	
* Description	

1/600

In the **Time Partition** area, enter the field code **dtime** and select the data type **TIMESTAMP**. After a table is published, data is written to the table based on the field added here.

Figure 4-77 Time Partition settings

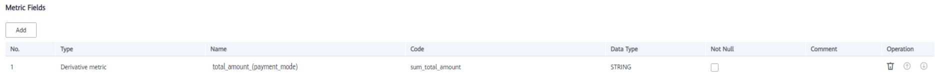


Time Partition

Code	Data Type
dtime	TIMESTAMP

In the **Metric Settings** area, 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 4-78 Metric settings



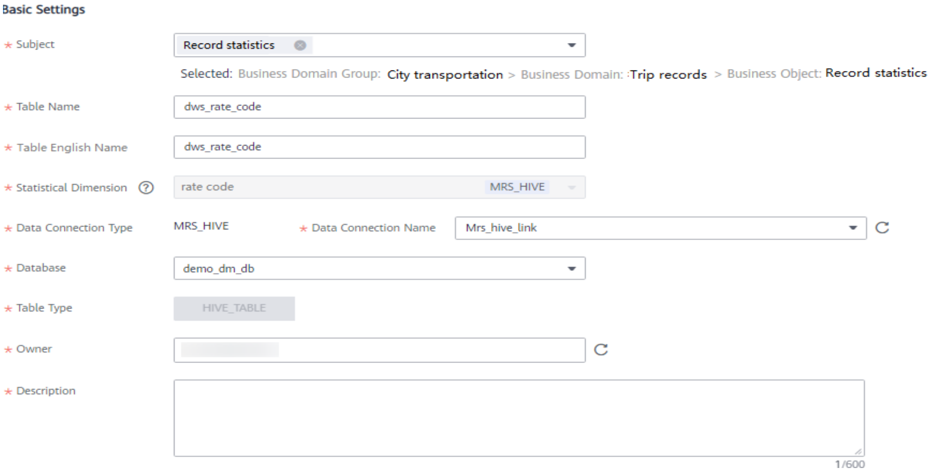
Metric Fields

No.	Type	Name	Code	Data Type	Not Null	Comment	Operation
1	Derivative metric	total_amount_(payment_mode)	sum_total_amount	STRING	□		

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 4-79 Creating a summary table named dws_rate_code (Basic Settings)

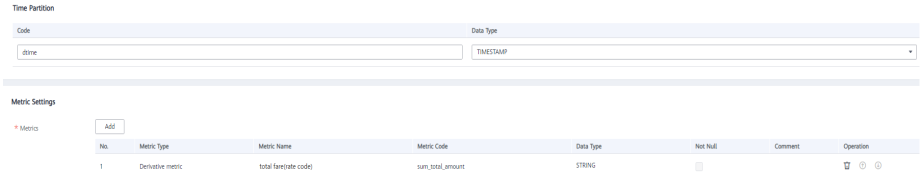


Basic Settings

* Subject	Record statistics
	Selected: Business Domain Group: City transportation > Business Domain: Trip records > Business Object: Record statistics
* Table Name	dws_rate_code
* Table English Name	dws_rate_code
* Statistical Dimension	rate code MRS_HIVE
* Data Connection Type	MRS_HIVE
* Data Connection Name	Mrs_hive_link
* Database	demo_dm_db
* Table Type	HIVE_TABLE
* Owner	[redacted]
* Description	

1/600

Figure 4-80 Creating a summary table named rate statistics (Metric Settings)



Time Partition

Code	Data Type
dtime	TIMESTAMP

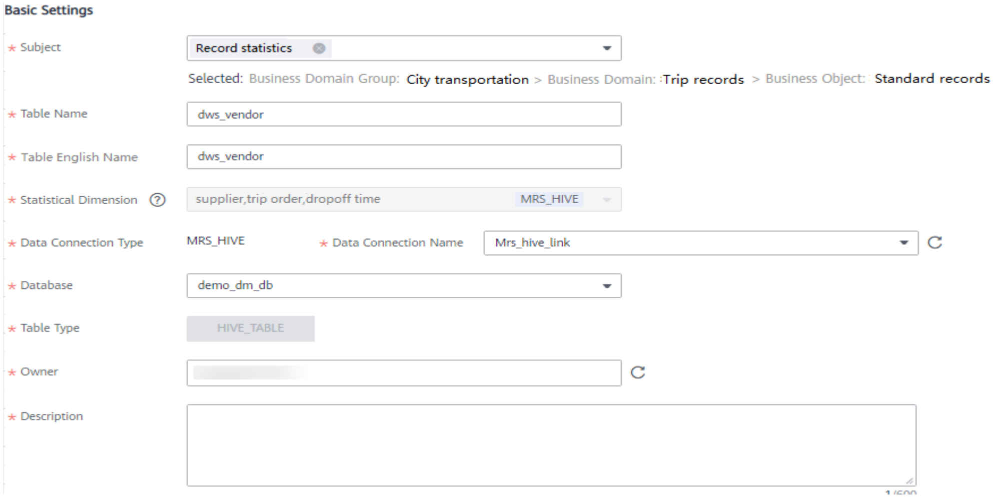
Metric Settings

* Metrics Add

No.	Metric Type	Metric Name	Metric Code	Data Type	Not Null	Comment	Operation
1	Derivative metric	total fare(rate code)	sum_total_amount	STRING	□		

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**.

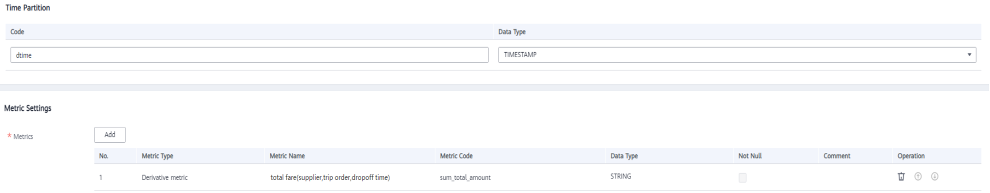
Figure 4-81 Creating a summary table named dws_vendor (Basic Settings)



Basic Settings

* Subject	Record statistics
	Selected: Business Domain Group: City transportation > Business Domain: Trip records > Business Object: Standard records
* Table Name	dws_vendor
* Table English Name	dws_vendor
* Statistical Dimension	supplier,trip order,dropoff time MRS_HIVE
* Data Connection Type	MRS_HIVE
* Data Connection Name	Mrs_hive_link
* Database	demo_dm_db
* Table Type	HIVE_TABLE
* Owner	[redacted]
* Description	

Figure 4-82 Creating a summary table named dws_vendor (Metric Settings)



Time Partition

Code	Data Type
dtime	TIMESTAMP

Metric Settings

* Metrics

No.	Metric Type	Metric Name	Metric Code	Data Type	Not Null	Comment	Operation
1	Derivative metric	total fare(supplier,trip order,dropoff time)	sum_total_amount	STRING	□		

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

Reviewing an Application

Step 1 Log in to the DataArts Studio console as a reviewer. Locate the created DataArts Studio instance and click **Access**. In the workspace list, locate the target workspace and click **DataArts Architecture**.

Step 2 Choose **Metrics** > **Review Center** in the left navigation pane, click the **Pending Review** tab, select the objects to be reviewed in the list, and click **Review** above the list.

Step 3 Enter review comments and click **Accept**.

----End

4.7 Step 6: DataArts Factory

DataArts Studio DataArts Factory provides a one-stop big data development environment and 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.

With DataArts Factory, you can perform a variety of operations such as data management, data integration, script development, job development, version management, job scheduling, O&M, and monitoring, facilitating data analysis and processing.

In the DataArts Factory module, perform the following steps:

1. **Managing data**
2. **Developing a Script**
3. **Developing a Batch Job**
   a. Use DataArts Migration to import historical data from OBS to the source data table of the SDI layer.
   b. Use an MRS Hive SQL script to cleanse the source data table and import the data to a standard business table at the DWI layer.
   c. Insert basic data into a dimension table.
   d. Import the standard business data at the DWI layer to a fact table at the DWR layer.
   e. Use Hive SQL to summarize data in the taxi trip order fact table and write the data into a summary table.
4. **O&M Scheduling**

Managing data

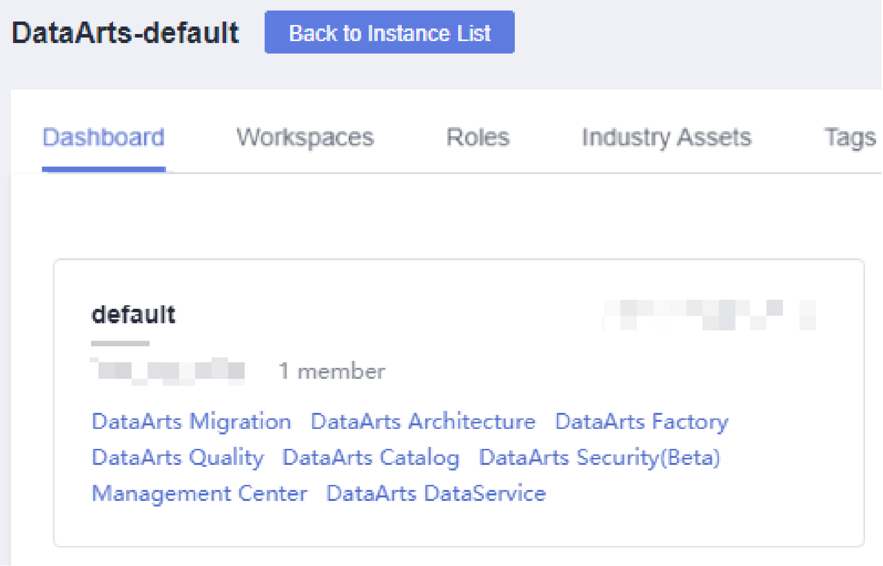
The data management function helps you quickly establish data models and provides you with data entities for script and job development. It includes creating data connections, creating databases, and creating data tables.

In this example, related data management operations have been performed in **Step 2: Preparations**.

Developing a Script

Step 1 Log in to the DataArts Studio console. Locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Factory**.

Figure 4-83 DataArts Factory

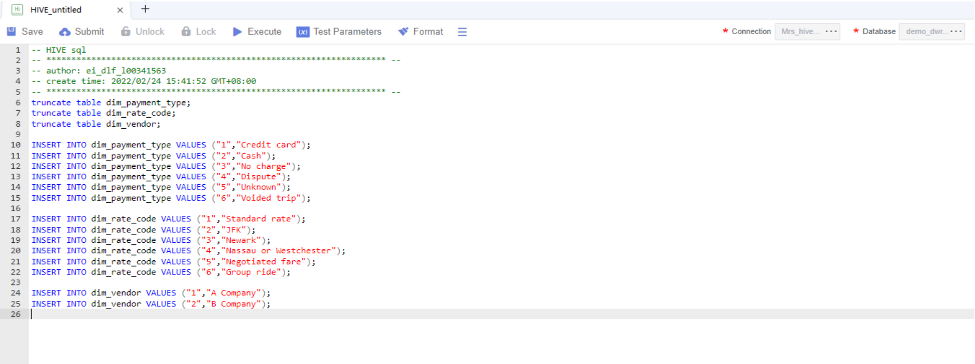


Step 2 In the navigation pane on the left, choose **Develop Script**. Right-click **Scripts** and choose **Create Directory** from the shortcut menu. In the dialog box displayed, enter the directory name, for example, **transport**, and click **OK**.

Step 3 In the script directory tree, right-click the **transport** directory and choose **Create Hive SQL Script** from the shortcut menu.

Step 4 In the created **HIVE_untitled** script, select **mrs_hive_link** for **Connection**, select **demo_dwr_db** for **Database**, and enter the script content.

Figure 4-84 Editing a script



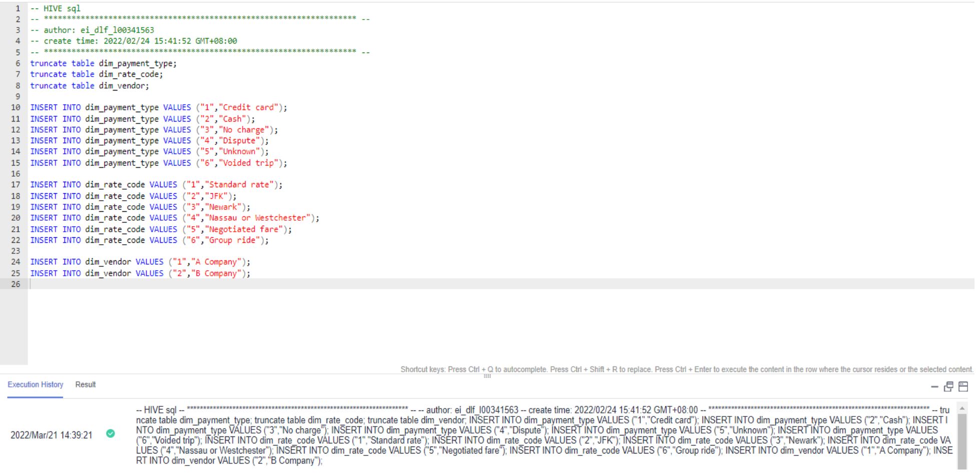
```
-- HIVE sql
-- *************************************************************** --
-- author: ei_dlf_100341563
-- create time: 2022/02/24 15:41:52 GMT+08:00
-- *************************************************************** --
truncate table dim_payment_type;
truncate table dim_rate_code;
truncate table dim_vendor;

INSERT INTO dim_payment_type VALUES ("1","Credit card");
INSERT INTO dim_payment_type VALUES ("2","Cash");
INSERT INTO dim_payment_type VALUES ("3","No charge");
INSERT INTO dim_payment_type VALUES ("4","Dispute");
INSERT INTO dim_payment_type VALUES ("5","Unknown");
INSERT INTO dim_payment_type VALUES ("6","Voided trip");

INSERT INTO dim_rate_code VALUES ("1","Standard rate");
INSERT INTO dim_rate_code VALUES ("2","JFK");
INSERT INTO dim_rate_code VALUES ("3","Newark");
INSERT INTO dim_rate_code VALUES ("4","Nassau or Westchester");
INSERT INTO dim_rate_code VALUES ("5","Negotiated fare");
INSERT INTO dim_rate_code VALUES ("6","Group ride");

INSERT INTO dim_vendor VALUES ("1","A Company");
INSERT INTO dim_vendor VALUES ("2","B Company");
```

This script is used to write the payment method, rate code, and vendor to the corresponding dimension table. The script content is as follows:

```
truncate table dim_payment_type;
truncate table dim_rate_code;
truncate table dim_vendor;

INSERT INTO dim_payment_type VALUES ("1","Credit card");
INSERT INTO dim_payment_type VALUES ("2","Cash");
INSERT INTO dim_payment_type VALUES ("3","No charge");
INSERT INTO dim_payment_type VALUES ("4","Dispute");
INSERT INTO dim_payment_type VALUES ("5","Unknown");
INSERT INTO dim_payment_type VALUES ("6","Voided trip");

INSERT INTO dim_rate_code VALUES ("1","Standard rate");
INSERT INTO dim_rate_code VALUES ("2","JFK");
INSERT INTO dim_rate_code VALUES ("3","Newark");
INSERT INTO dim_rate_code VALUES ("4","Nassau or Westchester");
INSERT INTO dim_rate_code VALUES ("5","Negotiated fare");
INSERT INTO dim_rate_code VALUES ("6","Group ride");

INSERT INTO dim_vendor VALUES ("1","A Company");
INSERT INTO dim_vendor VALUES ("2","B Company");
```

Step 5 Click **Execute** and check whether the script is correct.

Figure 4-85 Executing the script



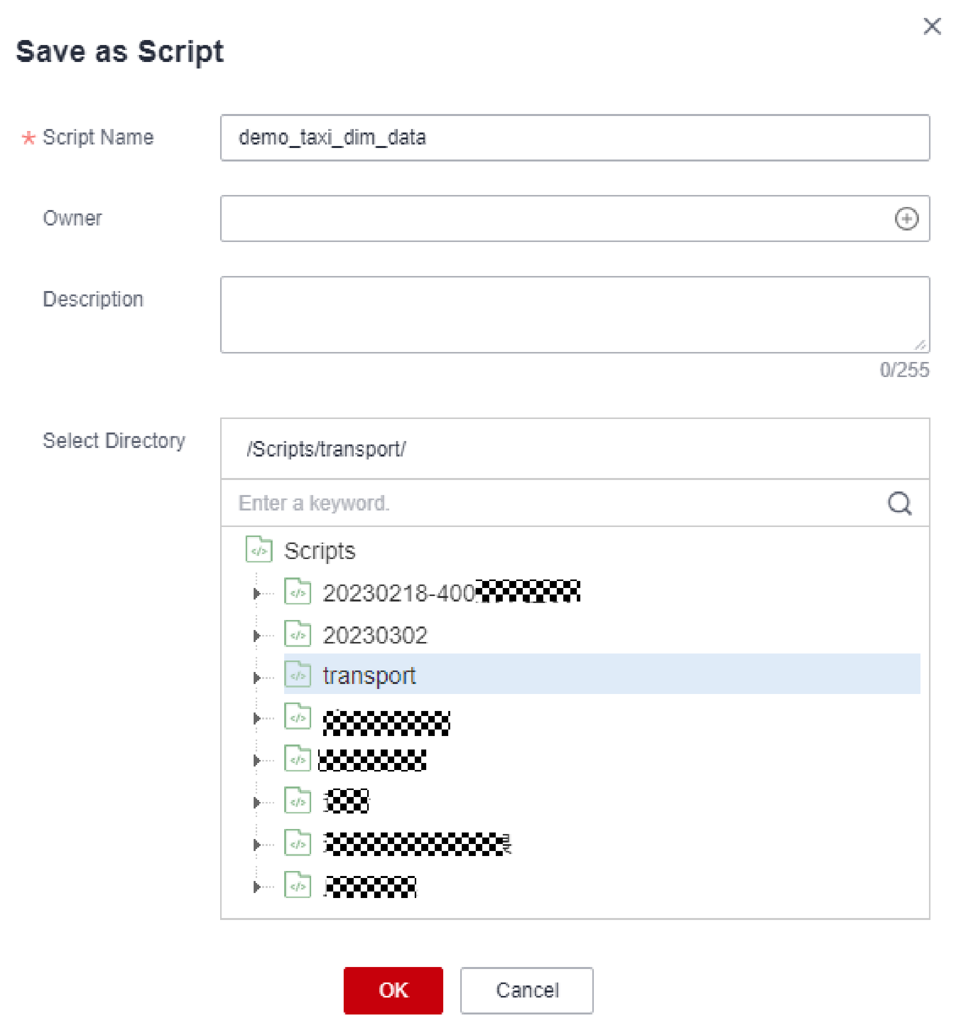
```
-- HIVE sql
-- ********************************************************************** --
-- author: ei_dlf_l00341563
-- create time: 2022/02/24 15:41:52 GMT+08:00
-- ********************************************************************** --
truncate table dim_payment_type;
truncate table dim_rate_code;
truncate table dim_vendor;

INSERT INTO dim_payment_type VALUES ("1","Credit card");
INSERT INTO dim_payment_type VALUES ("2","Cash");
INSERT INTO dim_payment_type VALUES ("3","No charge");
INSERT INTO dim_payment_type VALUES ("4","Dispute");
INSERT INTO dim_payment_type VALUES ("5","Unknown");
INSERT INTO dim_payment_type VALUES ("6","Voided trip");

INSERT INTO dim_rate_code VALUES ("1","Standard rate");
INSERT INTO dim_rate_code VALUES ("2","JFK");
INSERT INTO dim_rate_code VALUES ("3","Newark");
INSERT INTO dim_rate_code VALUES ("4","Nassau or Westchester");
INSERT INTO dim_rate_code VALUES ("5","Negotiated fare");
INSERT INTO dim_rate_code VALUES ("6","Group ride");

INSERT INTO dim_vendor VALUES ("1","A Company");
INSERT INTO dim_vendor VALUES ("2","B Company");
```

Shortcut keys: Press Ctrl + Q to autocomplete. Press Ctrl + Shift + R to replace. Press Ctrl + Enter to execute the content in the row where the cursor resides or the selected content.
Execution History
Result

```
2022/Mar/21 14:39:21

-- HIVE sql -- ********************************************************************** -- -- author: ei_dlf_l00341563 -- create time: 2022/02/24 15:41:52 GMT+08:00 -- ********************************************************************** -- tru
ncate table dim_payment_type; truncate table dim_rate_code; truncate table dim_vendor; INSERT INTO dim_payment_type VALUES ("1","Credit card"); INSERT INTO dim_payment_type VALUES ("2","Cash"); INSERT I
NTO dim_payment_type VALUES ("3","No charge"); INSERT INTO dim_payment_type VALUES ("4","Dispute"); INSERT INTO dim_payment_type VALUES ("5","Unknown"); INSERT INTO dim_payment_type VALUES
("6","Voided trip"); INSERT INTO dim_rate_code VALUES ("1","Standard rate"); INSERT INTO dim_rate_code VALUES ("2","JFK"); INSERT INTO dim_rate_code VALUES ("3","Newark"); INSERT INTO dim_rate_code VA
LUES ("4","Nassau or Westchester"); INSERT INTO dim_rate_code VALUES ("5","Negotiated fare"); INSERT INTO dim_rate_code VALUES ("6","Group ride"); INSERT INTO dim_vendor VALUES ("1","A Company"); INSE
RT INTO dim_vendor VALUES ("2","B Company");
```

Step 6 After the test is successful, click **Save**. In the displayed dialog box, enter the script name, for example, **demo_taxi_dim_data**, select the directory for saving the script, and click **OK**. Then click **Submit**.

Figure 4-86 Saving the script



Save as Script

* Script Name: demo_taxi_dim_data

Owner:

Description:

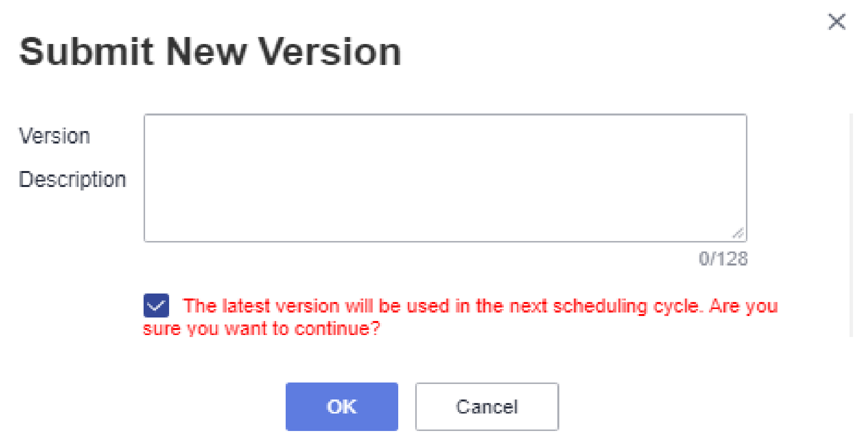
0/255

Select Directory: /Scripts/transport/

Enter a keyword.

- Scripts
  - 20230218-400[redacted]
  - 20230302
  - transport
  - [redacted]
  - [redacted]
  - [redacted]
  - [redacted]
  - [redacted]

Figure 4-87 Submitting the script version



Submit New Version

Version Description:

0/128

☑ The latest version will be used in the next scheduling cycle. Are you sure you want to continue?

Step 7 Repeat 4 to Step 6 to create the following scripts.

1. Script **demo_etl_sdi_dwi**: This script is used to write original data at the SDI layer to a standard business table at the DWI layer. The script content is as follows:

```
INSERT INTO
  demo_dwi_db.dwi_taxi_trip_data
SELECT
  `vendorid`,
  cast(
    regexp_replace(
      `tpep_pickup_datetime`,
      '(\\d{2})/(\\d{2})/(\\d{4}) (\\d{2}):(\\d{2}):(\\d{2}) (.*)',
      '$3-$1-$2 $4:$5:$6'
    ) as TIMESTAMP
  ),
  cast(
    regexp_replace(
      `tpep_dropoff_datetime`,
      '(\\d{2})/(\\d{2})/(\\d{4}) (\\d{2}):(\\d{2}):(\\d{2}) (.*)',
      '$3-$1-$2 $4:$5:$6'
    ) as TIMESTAMP
  ),
  `passenger_count`,
  `trip_distance`,
  `ratecodeid`,
  `store_fwd_flag`,
  `pulocationid`,
  `dolocationid`,
  `payment_type`,
  `fare_amount`,
  `extra`,
  `mta_tax`,
  `tip_amount`,
  `tolls_amount`,
  `improvement_surcharge`,
  `total_amount`
FROM
  demo_sdi_db.sdi_taxi_trip_data
WHERE
  trip_distance > 0
  and total_amount > 0
  and payment_type in (1, 2, 3, 4, 5, 6)
  and vendorid in (1, 2)
  and ratecodeid in (1, 2, 3, 4, 5, 6)
  and tpep_pickup_datetime < tpep_dropoff_datetime
  and tip_amount >= 0
  and fare_amount >= 0
  and extra >= 0
  and mta_tax >= 0
  and tolls_amount >= 0
  and improvement_surcharge >= 0
  and total_amount >= 0
  and (fare_amount + extra + mta_tax + tip_amount + tolls_amount + improvement_surcharge) = 
total_amount;
```

2. Script **demo_etl_dwi_dwr_fact**: This script is used to write the standard business data at the DWI layer to a fact table at the DWR layer. The script content is as follows:

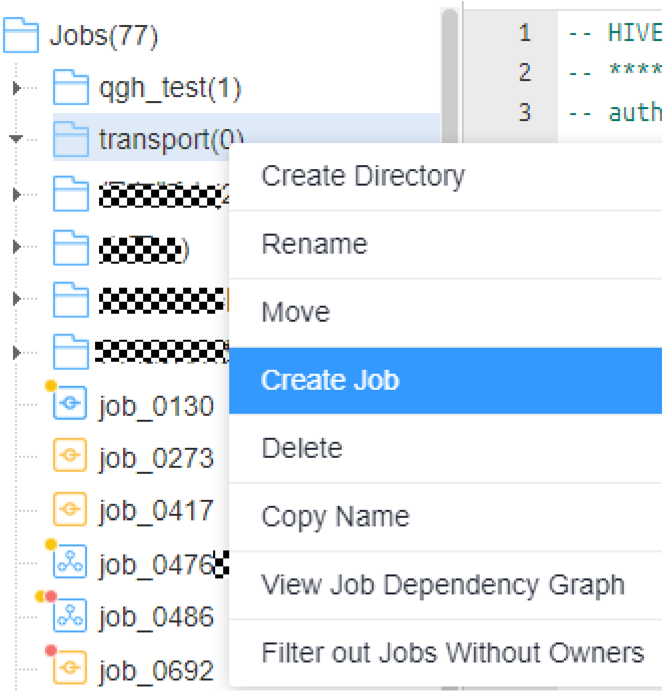
```
INSERT INTO
  demo_dwr_db.fact_stroke_order
SELECT
  rate_code_id,
  vendor_id,
  payment_type,
  tpep_dropoff_datetime,
  tpep_pickup_datetime,
  pu_location_id,
  do_location_id,
  fare_amount,
  extra,
  mta_tax,
  tip_amount,
  tolls_amount,
  improvement_surcharge,
  total_amount
FROM
  demo_dwi_db.dwi_taxi_trip_data;
```

----End

Developing a Batch Job

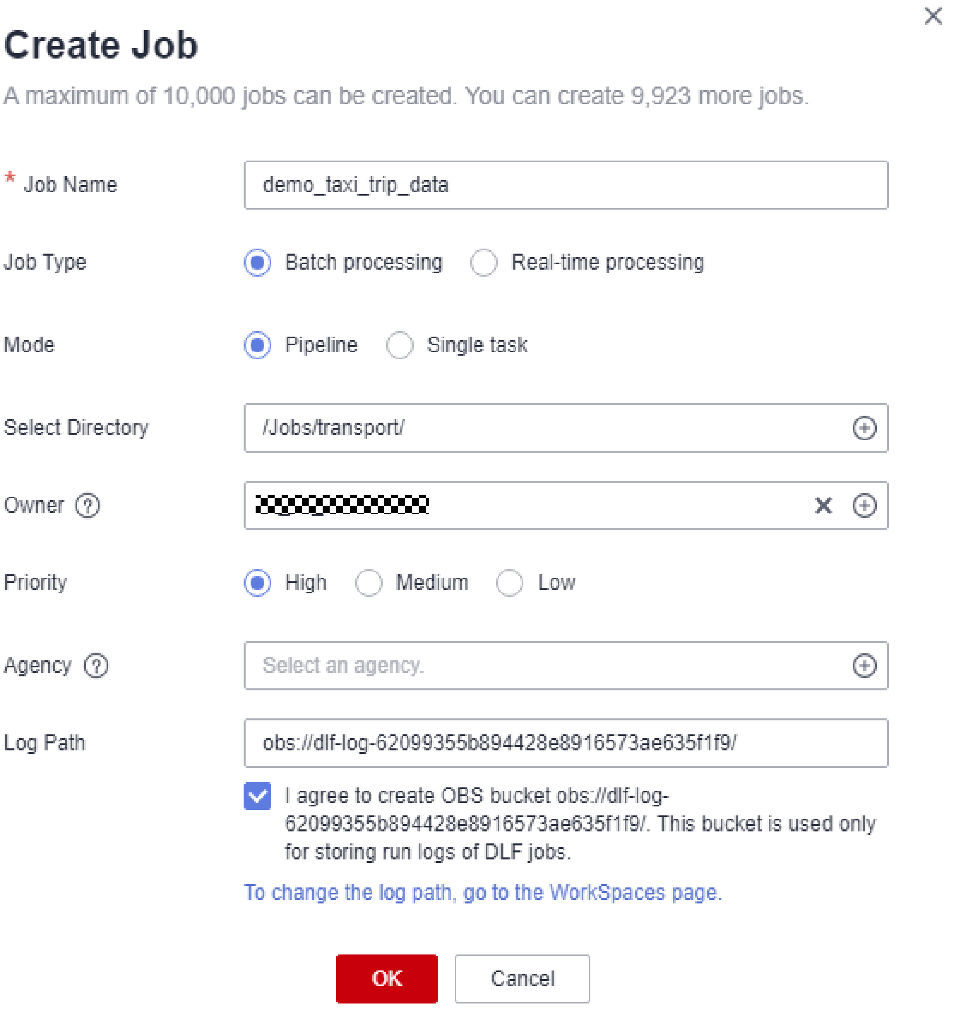
1. In the navigation pane of DataArts Studio DataArts Factory console, choose **Develop Job**. Right-click **Jobs** and choose **Create Directory** from the shortcut menu. In the directory tree, create a job directory as required, for example, **transport**.
2. Right-click the job directory and choose **Create Job** from the shortcut menu.

Figure 4-88 Creating a job



3. In the dialog box displayed, enter a job name, for example, **demo_taxi_trip_data**, set **Processing Mode** to **Batch processing**, retain the default values for other parameters, and click **OK**.

Figure 4-89 Creating a batch processing job



Create Job

A maximum of 10,000 jobs can be created. You can create 9,923 more jobs.

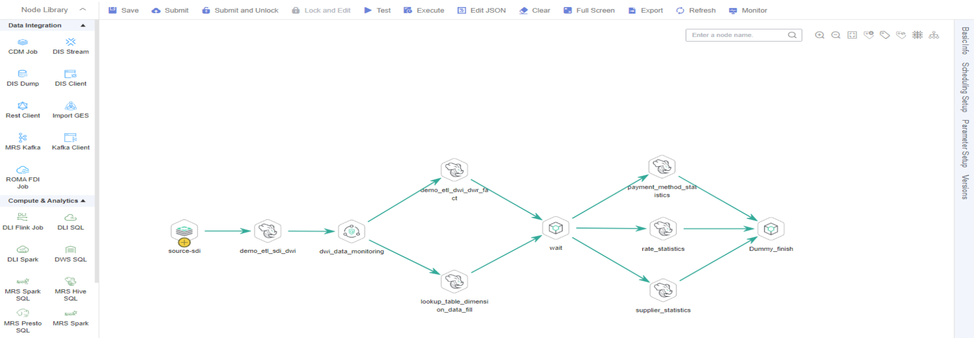
* Job Name	demo_taxi_trip_data
Job Type	● Batch processing ○ Real-time processing
Mode	● Pipeline ○ Single task
Select Directory	/Jobs/transport/
Owner	[redacted]
Priority	● High ○ Medium ○ Low
Agency	Select an agency.
Log Path	obs://dlf-log-62099355b894428e8916573ae635f1f9/

☑ I agree to create OBS bucket obs://dlf-log-62099355b894428e8916573ae635f1f9/. This bucket is used only for storing run logs of DLF jobs.

To change the log path, go to the WorkSpaces page.

4. Orchestrate a batch job, as shown in the figure below.

Figure 4-90 Orchestrating a job

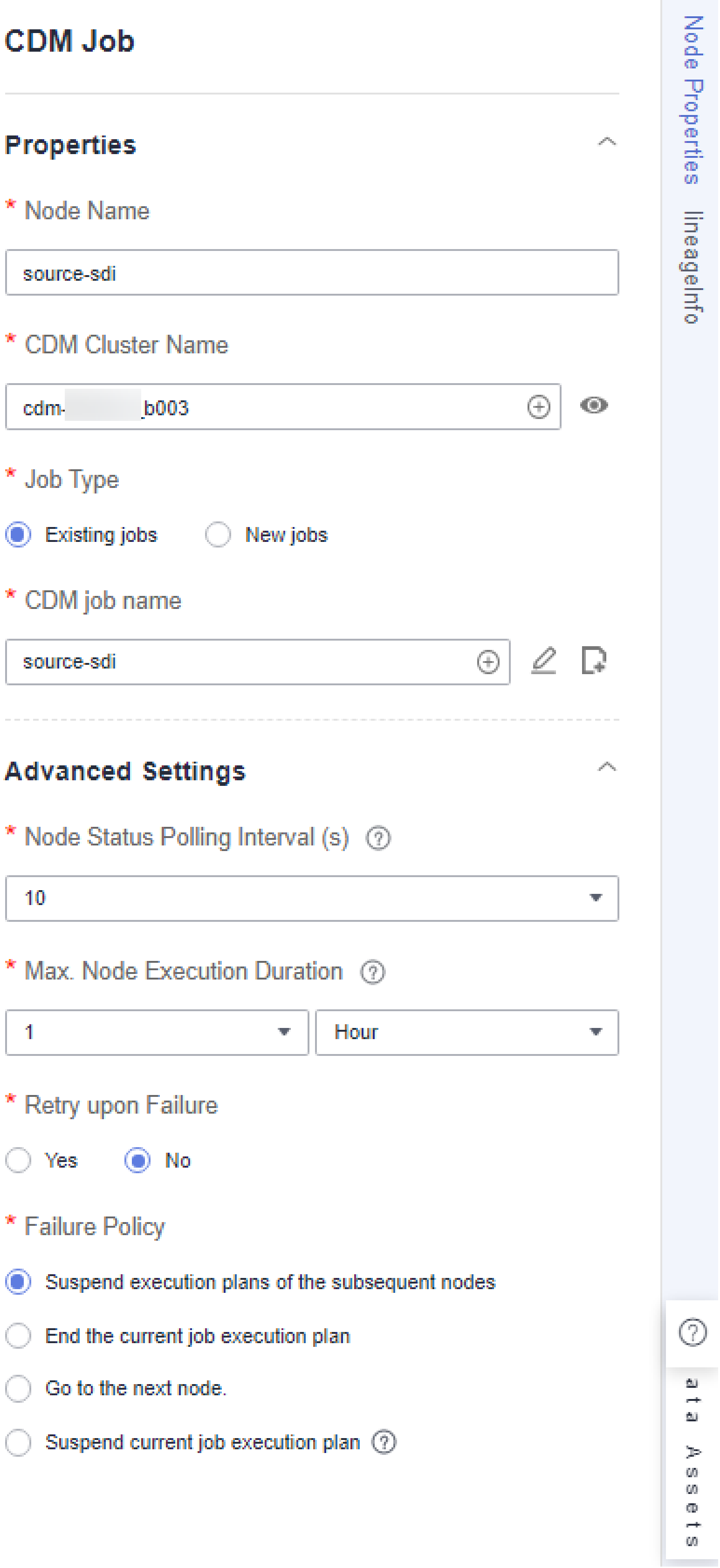


The node configurations are as follows:

- **source_sdi**: a CDM Job node, which is used to import data from OBS to the original table in MRS Hive. Set **CDM Cluster Name** and **CDM job**

name to the cluster and job created in **Step 3: DataArts Migration**, respectively. (The following figure shows an example.)

Figure 4-91 source_sdi node properties



CDM Job

Properties

* Node Name

source-sdi

* CDM Cluster Name

cdm-[redacted]b003

* Job Type

● Existing jobs ○ New jobs

* CDM job name

source-sdi

Advanced Settings

* Node Status Polling Interval (s)

10

* Max. Node Execution Duration

1 Hour

* Retry upon Failure

○ Yes ● No

* Failure Policy

● Suspend execution plans of the subsequent nodes

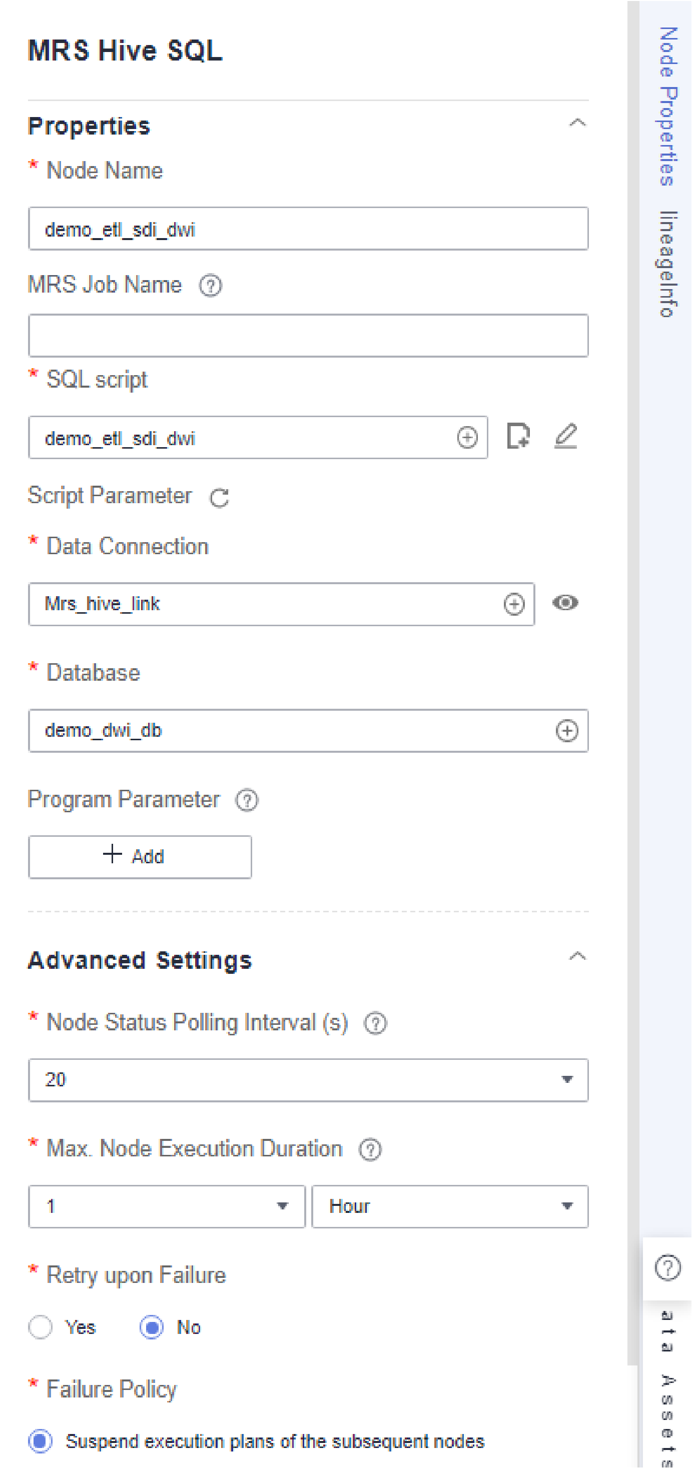
○ End the current job execution plan

○ Go to the next node.

○ Suspend current job execution plan

– **demo_etl_sdi_dwi**: an MRS Hive SQL node, which is used to cleanse and filter data in an original table at the SDI layer and write valid data into the standard business table **dwi_taxi_trip_data** at the DWI layer in DataArts Architecture. Set **SQL script** to the **demo_etl_sdi_dwi** script created in **Developing a Script**.

Figure 4-92 demo_etl_sdi_dwi node properties



MRS Hive SQL

Properties

* Node Name: demo_etl_sdi_dwi

MRS Job Name

* SQL script: demo_etl_sdi_dwi

Script Parameter

* Data Connection: Mrs_hive_link

* Database: demo_dwi_db

Program Parameter

Advanced Settings

* Node Status Polling Interval (s): 20

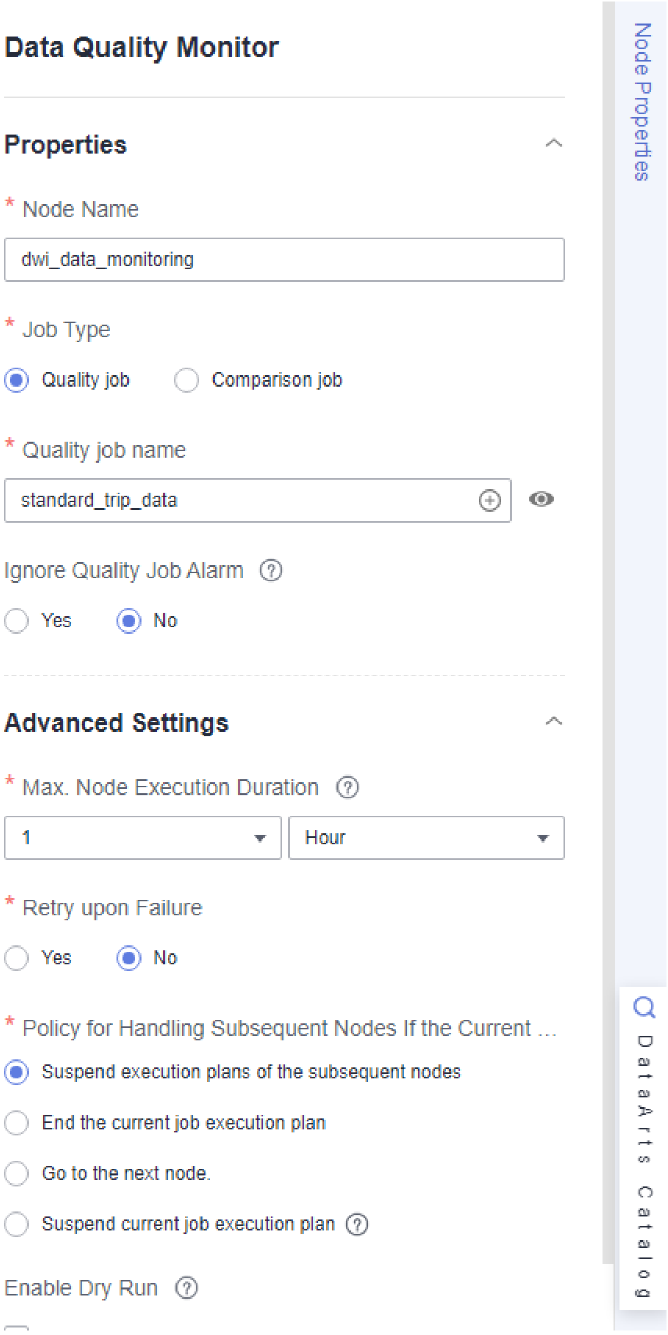
* Max. Node Execution Duration: 1 Hour

* Retry upon Failure: ○ Yes ● No

* Failure Policy: ● Suspend execution plans of the subsequent nodes

– **dwi_data_monitoring**: a Data Quality Monitor node, which is used to monitor the quality of standard business data at the DWI layer. Set **Quality Rule Name** to **standard business data**, which is automatically generated when the standard business table at the DWI layer is published.

Figure 4-93 dwi_data_monitoring node



Data Quality Monitor

Properties

* Node Name: dwi_data_monitoring

* Job Type: ● Quality job ○ Comparison job

* Quality job name: standard_trip_data

Ignore Quality Job Alarm: ○ Yes ● No

Advanced Settings

* Max. Node Execution Duration: 1 Hour

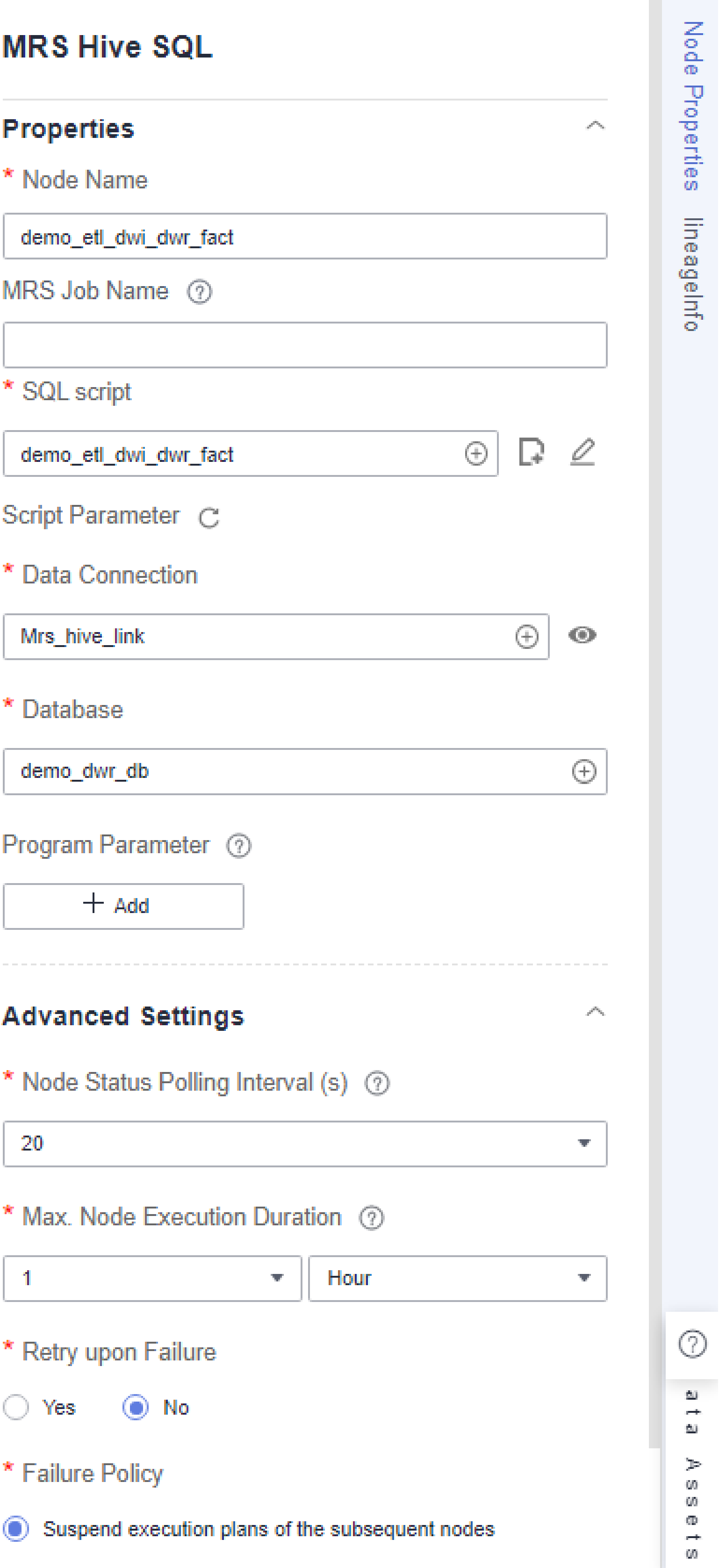
* Retry upon Failure: ○ Yes ● No

* Policy for Handling Subsequent Nodes If the Current ...
● Suspend execution plans of the subsequent nodes
○ End the current job execution plan
○ Go to the next node.
○ Suspend current job execution plan

Enable Dry Run

– **demo_etl_dwi_dwr_fact**: an MRS Hive SQL node, which is used to write source data at the DWI layer to fact table **fact_stroke_order** at the DWR layer. Set **SQL script** to the **demo_etl_dwi_dwr_fact** script created in **Developing a Script**.

Figure 4-94 demo_etl_dwi_dwr_fact node properties



MRS Hive SQL

Properties

* Node Name

demo_etl_dwi_dwr_fact

MRS Job Name

* SQL script

demo_etl_dwi_dwr_fact

Script Parameter

* Data Connection

Mrs_hive_link

* Database

demo_dwr_db

Program Parameter

Advanced Settings

* Node Status Polling Interval (s)

20

* Max. Node Execution Duration

1 Hour

* Retry upon Failure

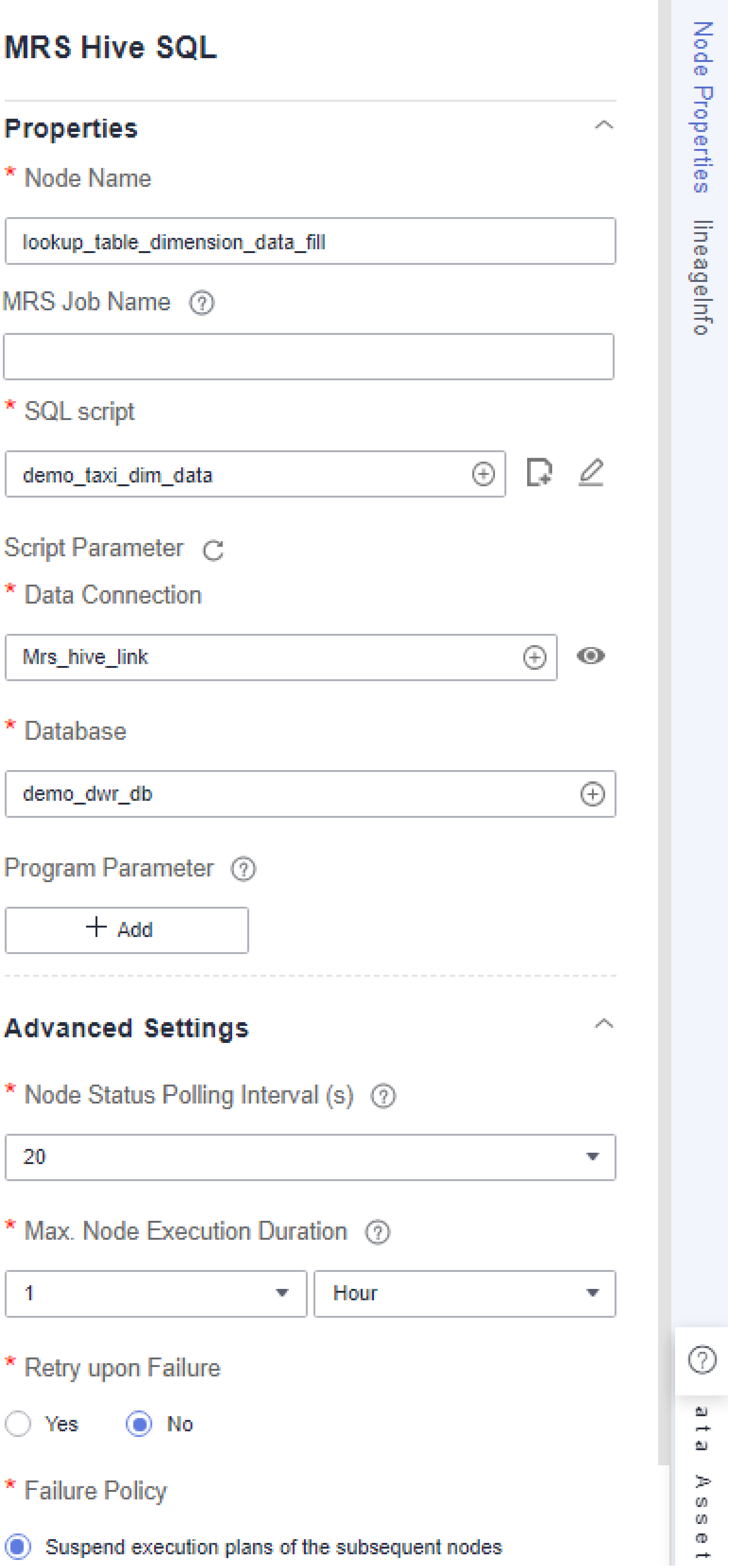
○ Yes ● No

* Failure Policy

● Suspend execution plans of the subsequent nodes

- **demo_lookup_dimension_dwr**: an MRS Hive SQL node, which is used to write the payment type, rate code, and vendor information to the corresponding dimension table at the DWR layer. Set **SQL script** to the **demo_taxi_dim_data** script created in **Developing a Script**.

Figure 4-95 demo_lookup_dimension_dwr node properties



MRS Hive SQL

Properties

* Node Name

lookup_table_dimension_data_fill

MRS Job Name

* SQL script

demo_taxi_dim_data

Script Parameter

* Data Connection

Mrs_hive_link

* Database

demo_dwr_db

Program Parameter

Advanced Settings

* Node Status Polling Interval (s)

20

* Max. Node Execution Duration

1 Hour

* Retry upon Failure

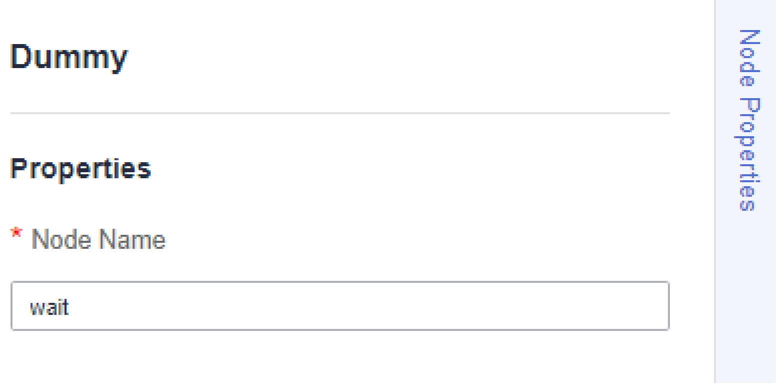
○ Yes ● No

* Failure Policy

● Suspend execution plans of the subsequent nodes

- **dummy_pending**: a Dummy node, which does not perform any operation but waits until the previous node stops running.

Figure 4-96 dummy_pending node



Dummy

Properties

* Node Name

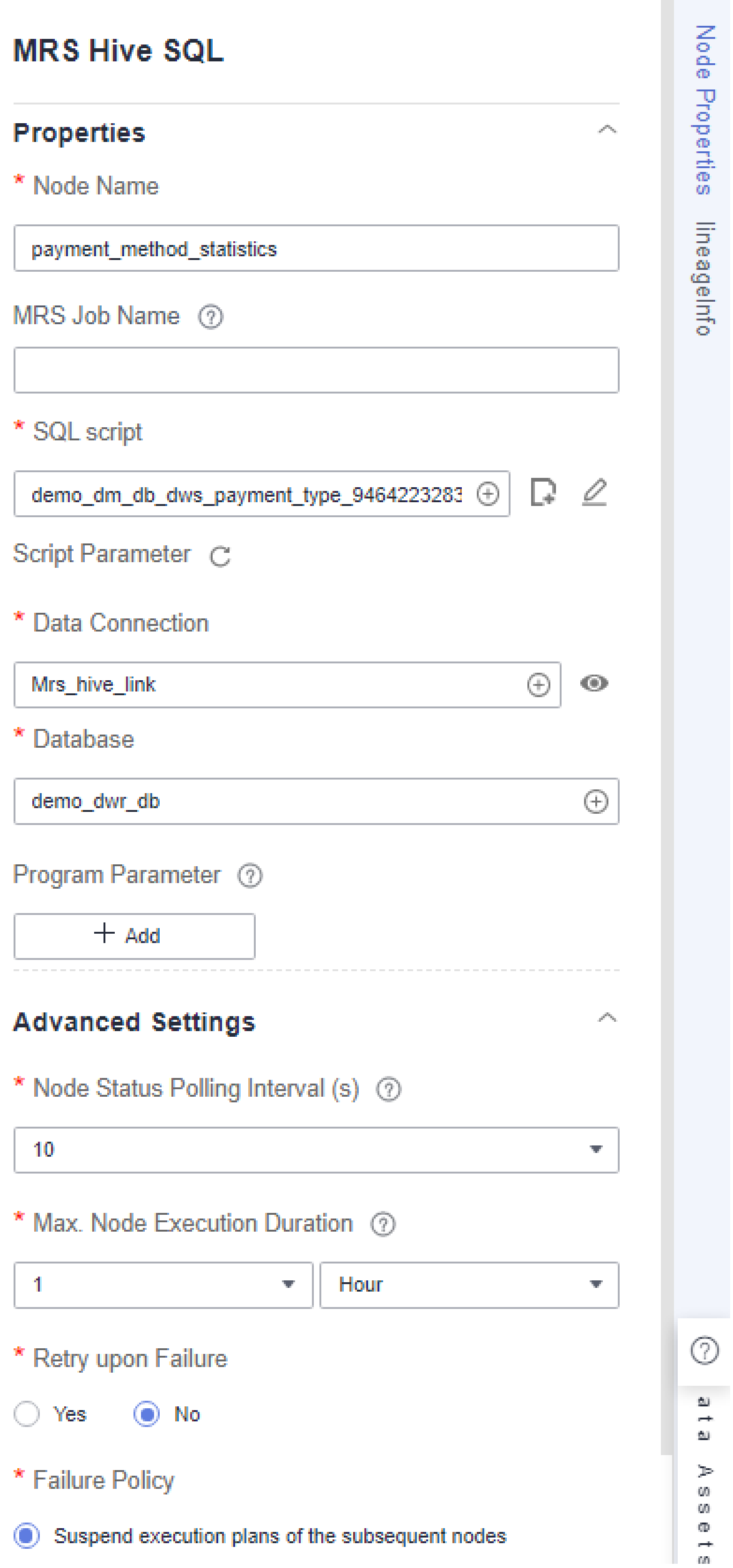
wait

- **summary_by_payment_type**: an MRS Hive SQL node, which collects statistics on the total revenue till the current date by payment type. This node is a data development job automatically generated when summary table **summary_by_payment_type** is published. The job name is prefixed with **demo_dm_db_dws_payment_type_** and followed by *Database name_Summary table code*. After the node is copied, set **Data Connection** and **Database** for the node. You must set **Database** to the database where the fact table is located.

NOTE

To enable a data development job to be automatically generated, ensure that you have selected **Create data development jobs** in **Configuration Center Management**.

Figure 4-97 summary_by_payment_type node properties



MRS Hive SQL

Properties

* Node Name

payment_method_statistics

MRS Job Name

* SQL script

demo_dm_db_dws_payment_type_9464223283

Script Parameter

* Data Connection

Mrs_hive_link

* Database

demo_dwr_db

Program Parameter

+ Add

Advanced Settings

* Node Status Polling Interval (s)

10

* Max. Node Execution Duration

1 Hour

* Retry upon Failure

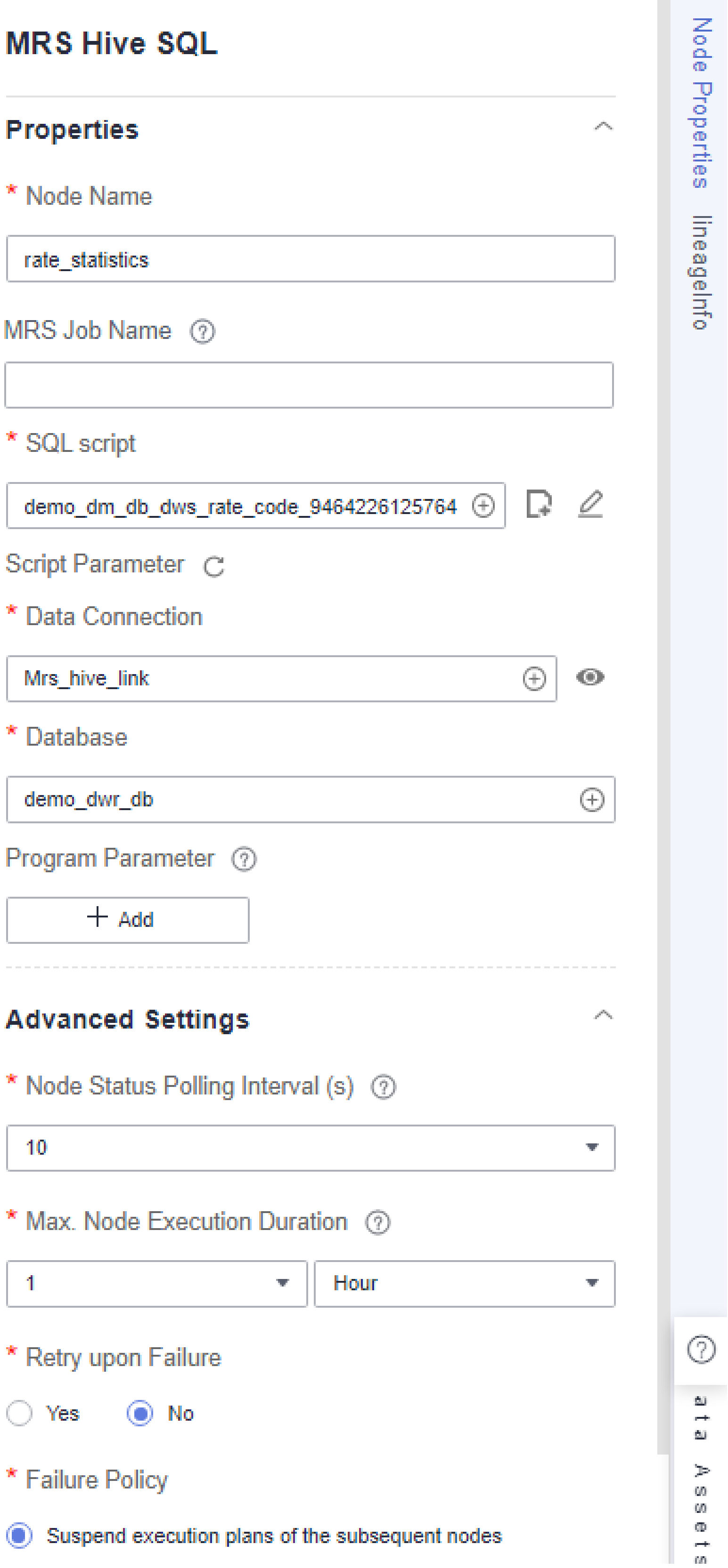
○ Yes ● No

* Failure Policy

● Suspend execution plans of the subsequent nodes

- **summary_by_rate_code**: an MRS Hive SQL node, which collects statistics on the total revenue till the current date by rate code. This node is a data development job automatically generated when summary table **summary_by_rate_code** is published. The job name is prefixed with **demo_dm_db_dws_rate_code_** and followed by *Database name_Summary table code*. After the node is copied, set **Data Connection** and **Database** for the node. You must set **Database** to the database where the fact table is located.

Figure 4-98 summary_by_rate_code node properties



MRS Hive SQL

Properties

* Node Name

rate_statistics

MRS Job Name

* SQL script

demo_dm_db_dws_rate_code_9464226125764

Script Parameter

* Data Connection

Mrs_hive_link

* Database

demo_dwr_db

Program Parameter

+ Add

Advanced Settings

* Node Status Polling Interval (s)

10

* Max. Node Execution Duration

1 Hour

* Retry upon Failure

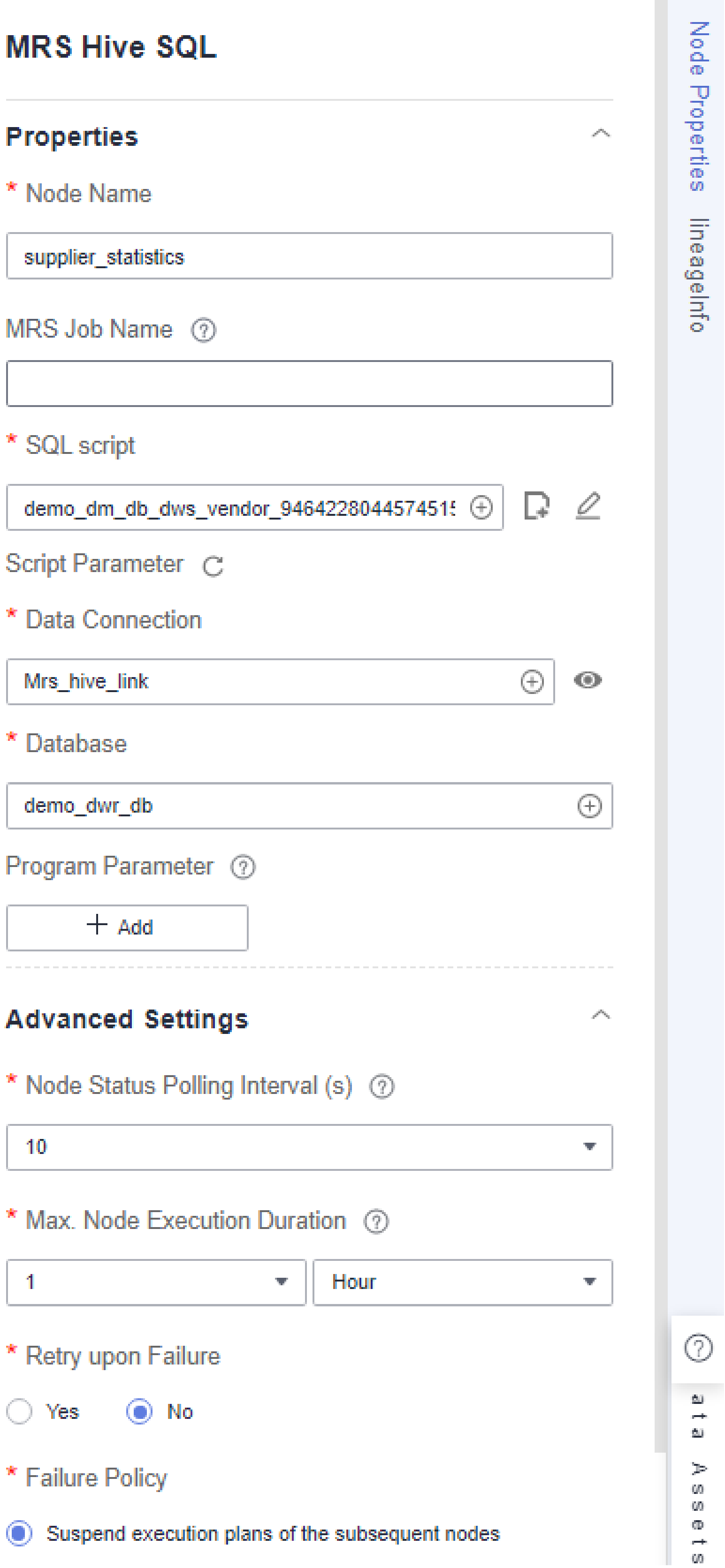
○ Yes ● No

* Failure Policy

● Suspend execution plans of the subsequent nodes

- **summary_by_vendor**: an MRS Hive SQL node, which collects statistics on the total revenue of each time dimension till the current date by vendor. This node is a data development job automatically generated when summary table **summary_by_vendor** is published. The job name is prefixed with **demo_dm_db_dws_vendor_** and followed by **Database name_Summary table code**. After the node is copied, set **Data Connection** and **Database** for the node. You must set **Database** to the database where the fact table is located.

Figure 4-99 summary_by_vendor node properties



MRS Hive SQL

Properties

* Node Name

supplier_statistics

MRS Job Name

* SQL script

demo_dm_db_dws_vendor_946422804457451!

Script Parameter

* Data Connection

Mrs_hive_link

* Database

demo_dwr_db

Program Parameter

+ Add

Advanced Settings

* Node Status Polling Interval (s)

10

* Max. Node Execution Duration

1 Hour

* Retry upon Failure

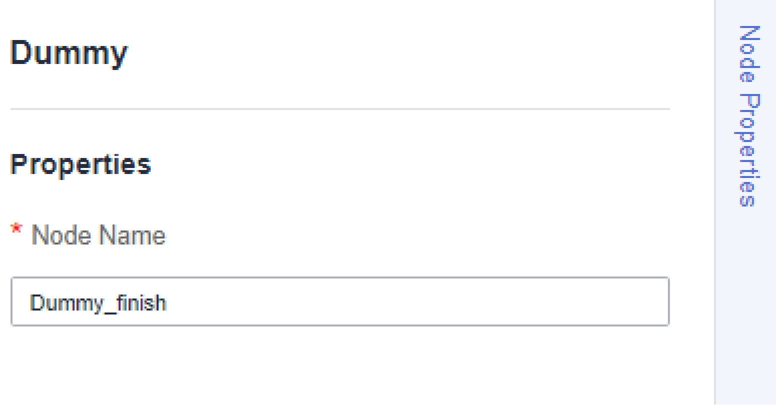
○ Yes ● No

* Failure Policy

● Suspend execution plans of the subsequent nodes

– **Dummy_finish**: a Dummy node, which marks the end of a job.

Figure 4-100 Dummy_finish node



Dummy

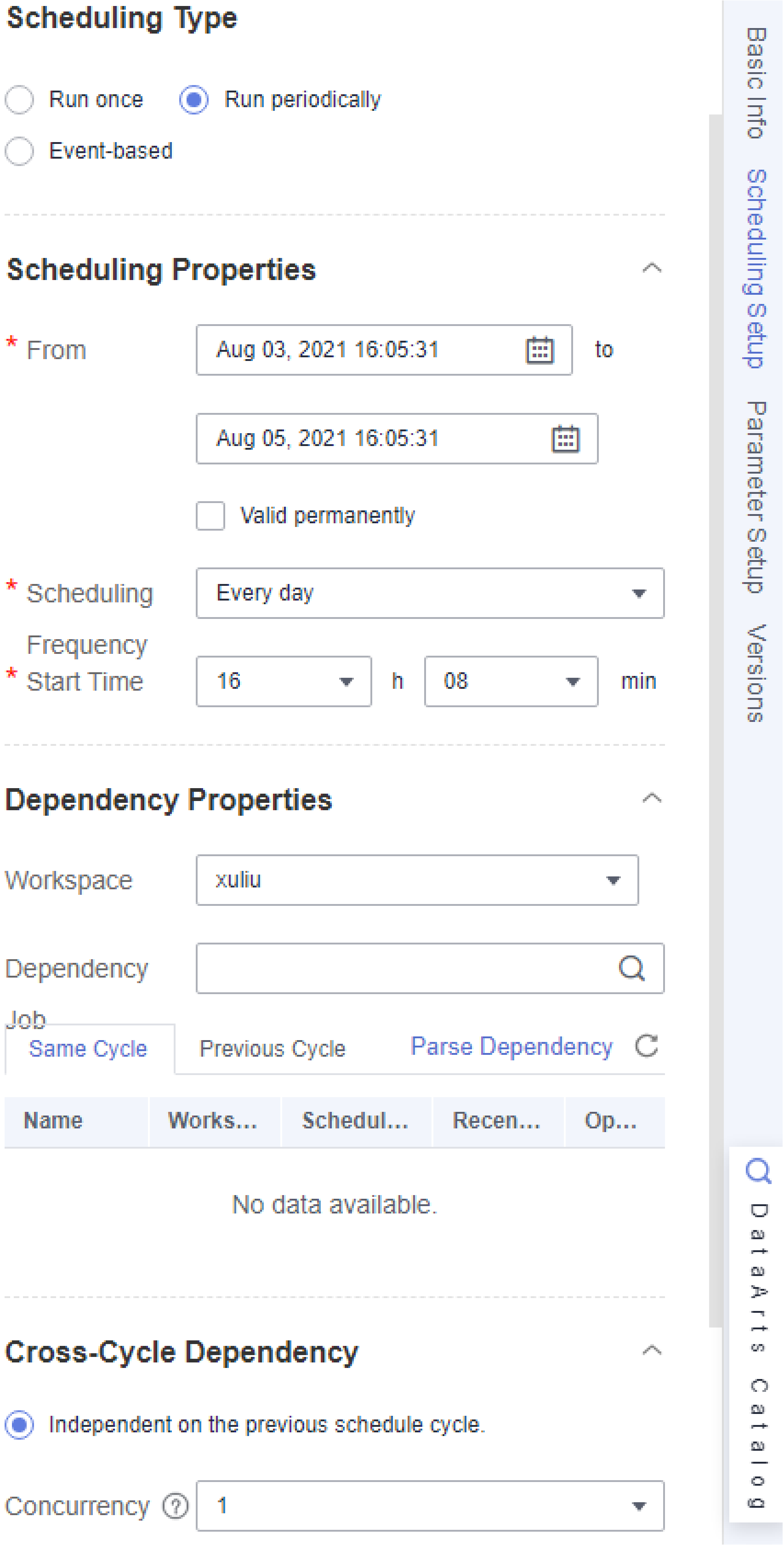
Properties

* Node Name

Dummy_finish

5. After the job orchestration is complete, check whether the job orchestration is correct by clicking **Test**.
6. Configure the job scheduling mode as required. Click **Scheduling Setup** in the right pane. Currently, three scheduling types are available: **Run once**, **Run periodically**, and **Event-based**.

Figure 4-101 Configuring the job scheduling mode



Scheduling Type

○ Run once ● Run periodically
○ Event-based

Scheduling Properties

* From: Aug 03, 2021 16:05:31 to Aug 05, 2021 16:05:31

□ Valid permanently

* Scheduling Frequency: Every day

* Start Time: 16 h 08 min

Dependency Properties

Workspace: xuliu

Dependency Job:

Name	Works...	Schedul...	Recen...	Op...

No data available.

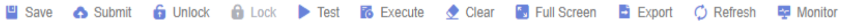
Cross-Cycle Dependency

● Independent on the previous schedule cycle.

Concurrency: 1

7. After configuring the scheduling parameters, click **Save** to save the job and click **Submit** to submit the job version. Then, click **Execute** to start job scheduling.

Figure 4-102 Saving, submitting, and executing the job

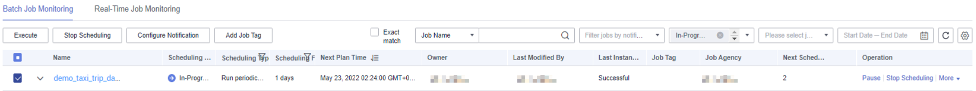


O&M Scheduling

You can use the O&M scheduling function to view the running statuses of jobs and job instances.

1. In the left navigation pane of DataArts Factory, choose **Monitoring** > **Monitor Job**.
2. Click the **Batch Job Monitoring** tab.
3. On this page, you can view the scheduling start time, frequency, and status of batch jobs. Select jobs and click **Execute**, **Stop Scheduling**, or **Configure Notification** to perform operations on the jobs.

Figure 4-103 Processing jobs in batches

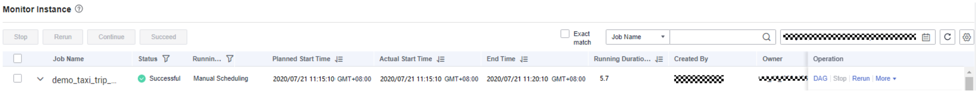


☑	Name	Scheduling ...	Scheduling Typ	Scheduling F	Next Plan Time	Owner	Last Modified By	Last Instan...	Job Tag	Job Agency	Next Sched...	Operation
☑	demo_taxi_trip_da...	In-Progr...	Run periodic...	1 days	May 23, 2022 02:24:00 GMT+0...	[redacted]	[redacted]	Successful		[redacted]	2	Pause | Stop Scheduling | More

4. In the left navigation pane, choose **Monitoring** > **Monitor Instance**.

   On the **Monitor Instance** page, you can view the running details and logs of job instances.

Figure 4-104 Monitoring instances

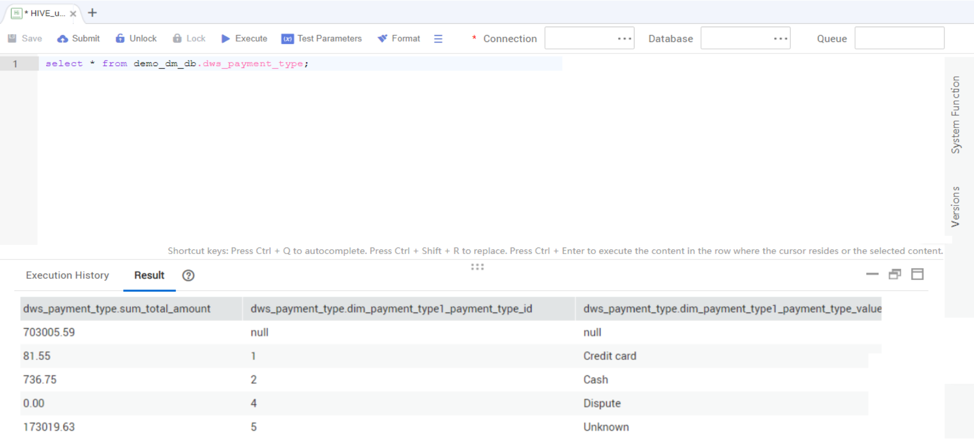


□	Job Name	Status	Runnin...	Planned Start Time	Actual Start Time	End Time	Running Duratio...	Created By	Owner	Operation
□	demo_taxi_trip_...	Successful	Manual Scheduling	2020/07/21 11:15:10 GMT+08:00	2020/07/21 11:15:10 GMT+08:00	2020/07/21 11:20:10 GMT+08:00	5.7	[redacted]	[redacted]	DAG | Stop | Rerun | More

5. After the job is successfully executed, you can preview the data in the summary table on the DataArts Studio **DataArts Catalog** page. For details, see **Step 8: DataArts Catalog**. You can also create a Hive SQL script on the **Develop Script** page of **DataArts Factory** and run the following command to query the result. If the execution is successful, a result similar to the following is displayed:

```
SELECT * FROM demo_dm_db.dws_payment_type;
```

Figure 4-105 Querying results



```
select * from demo_dm_db.dws_payment_type;
```

Shortcut keys: Press Ctrl + Q to autocomplete. Press Ctrl + Shift + R to replace. Press Ctrl + Enter to execute the content in the row where the cursor resides or the selected content.

dws_payment_type.sum_total_amount	dws_payment_type.dim_payment_type1_payment_type_id	dws_payment_type.dim_payment_type1_payment_type_value
703005.59	null	null
81.55	1	Credit card
736.75	2	Cash
0.00	4	Dispute
173019.63	5	Unknown

4.8 Step 7: DataArts Quality

DataArts Quality allows you to manage the quality of data in the 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.

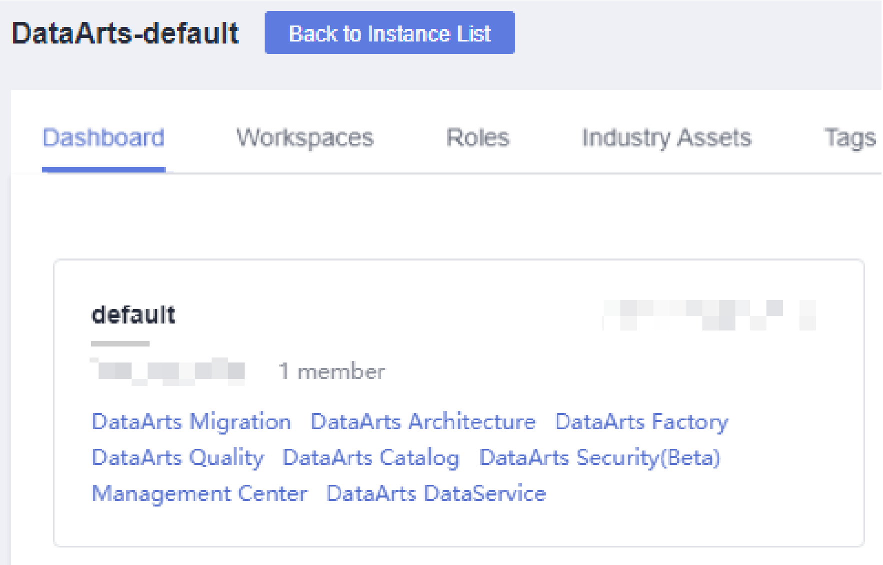
In the Data Quality module, the quality of metrics and data can be monitored.

Viewing Quality Jobs

After a job is executed during data development, you can view the running result of the job on the **DataArts Quality** page.

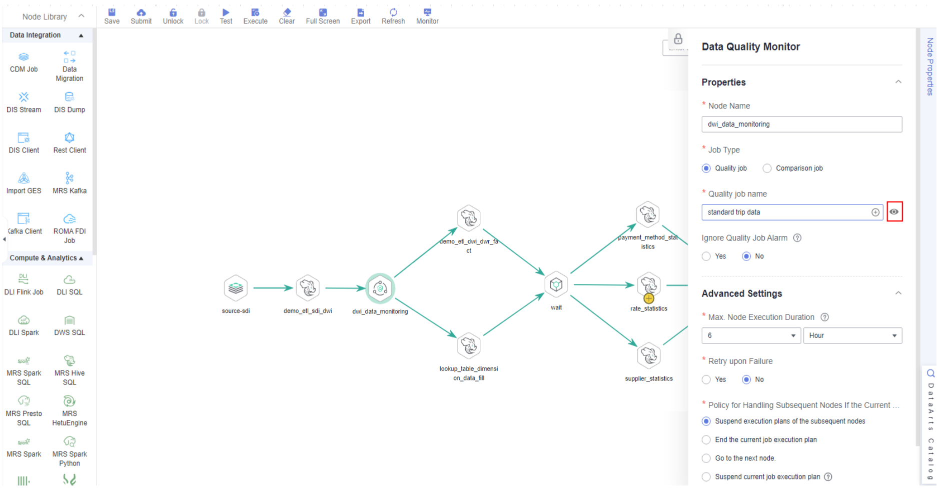
Step 1 Log in to the DataArts Studio console. Locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Factory**.

Figure 4-106 DataArts Factory



Step 2 On the **Develop Job** page under **DataArts Factory**, open the job created in **Step 6: DataArts Factory**, and click the data quality monitor node in the job. In **Node Properties**, click 👁 next to **Quality Rule Name** to display the **Quality Jobs** page under **DataArts Quality**.

Figure 4-107 Quality job node



Data Quality Monitor

Properties

* Node Name: dwi_data_monitoring

* Job Type: ● Quality job ○ Comparison job

* Quality job name: standard trip data

Ignore Quality Job Alarm: ○ Yes ● No

Advanced Settings

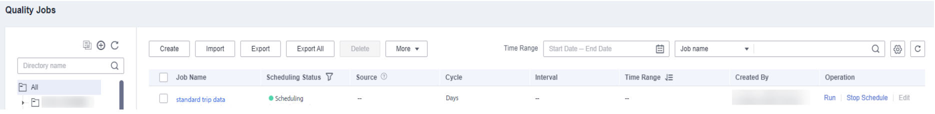
* Max. Node Execution Duration: 6 Hour

* Retry upon Failure: ○ Yes ● No

* Policy for Handling Subsequent Nodes If the Current ...
● Suspend execution plans of the subsequent nodes
○ End the current job execution plan
○ Go to the next node.
○ Suspend current job execution plan

Step 3 Click the name of a quality job to view its basic configuration.

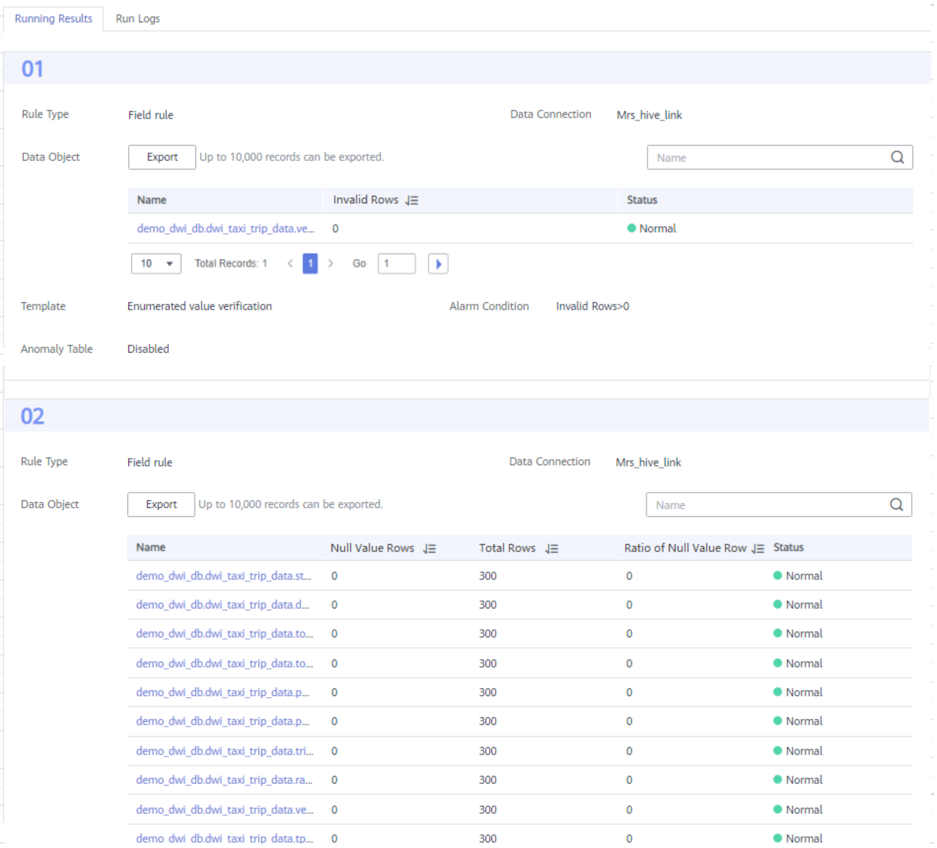
Figure 4-108 Quality job list



Job Name	Scheduling Status	Source	Cycle	Interval	Time Range	Created By	Operation
standard trip data	● Scheduling	--	Days	--	--	[redacted]	Run | Stop Schedule | Edit

Step 4 In the left navigation pane, choose **O&M**. In the right pane, click **Details** in the **Operation** column to view the running result of the quality job.

Figure 4-109 Running result of the quality job



01

Rule Type: Field rule
Data Connection: Mrs_hive_link

Data Object: Export Up to 10,000 records can be exported.

Name	Invalid Rows	Status
demo_dwi_db.dwi_taxi_trip_data.ve...	0	● Normal

10 Total Records: 1 1 Go 1

Template: Enumerated value verification
Alarm Condition: Invalid Rows>0

Anomaly Table: Disabled

02

Rule Type: Field rule
Data Connection: Mrs_hive_link

Data Object: Export Up to 10,000 records can be exported.

Name	Null Value Rows	Total Rows	Ratio of Null Value Row	Status
demo_dwi_db.dwi_taxi_trip_data.st...	0	300	0	● Normal
demo_dwi_db.dwi_taxi_trip_data.d...	0	300	0	● Normal
demo_dwi_db.dwi_taxi_trip_data.to...	0	300	0	● Normal
demo_dwi_db.dwi_taxi_trip_data.to...	0	300	0	● Normal
demo_dwi_db.dwi_taxi_trip_data.p...	0	300	0	● Normal
demo_dwi_db.dwi_taxi_trip_data.p...	0	300	0	● Normal
demo_dwi_db.dwi_taxi_trip_data.tri...	0	300	0	● Normal
demo_dwi_db.dwi_taxi_trip_data.ra...	0	300	0	● Normal
demo_dwi_db.dwi_taxi_trip_data.ve...	0	300	0	● Normal
demo_dwi_db.dwi_taxi_trip_data.tp...	0	300	0	● Normal

----End

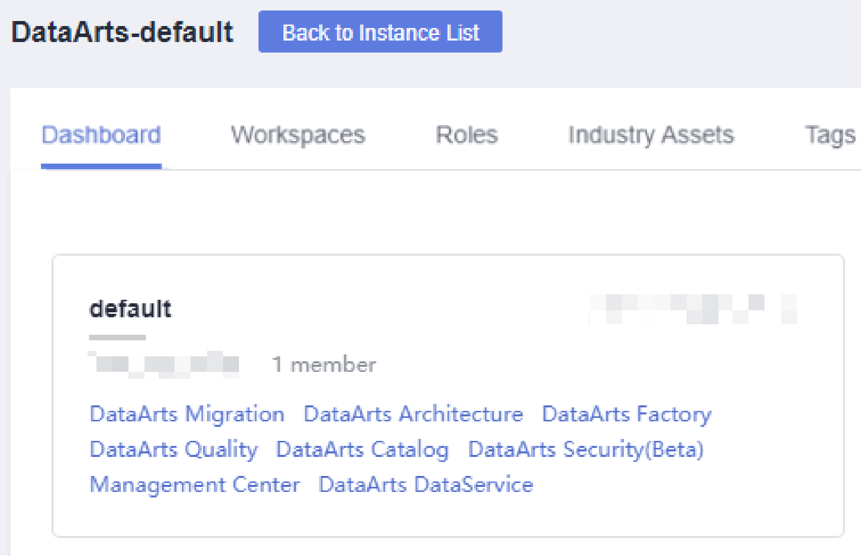
Monitoring Business Metrics

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. In this example, the system monitors the revenue of a taxi on a day and generates an alarm if the revenue is less than 500. The procedure 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 Quality**.

Figure 4-110 DataArts Quality



DataArts-default

Back to Instance List

Dashboard | Workspaces | Roles | Industry Assets | Tags

default [redacted]

[redacted] 1 member

DataArts Migration DataArts Architecture DataArts Factory
DataArts Quality DataArts Catalog DataArts Security(Beta)
Management Center DataArts DataService

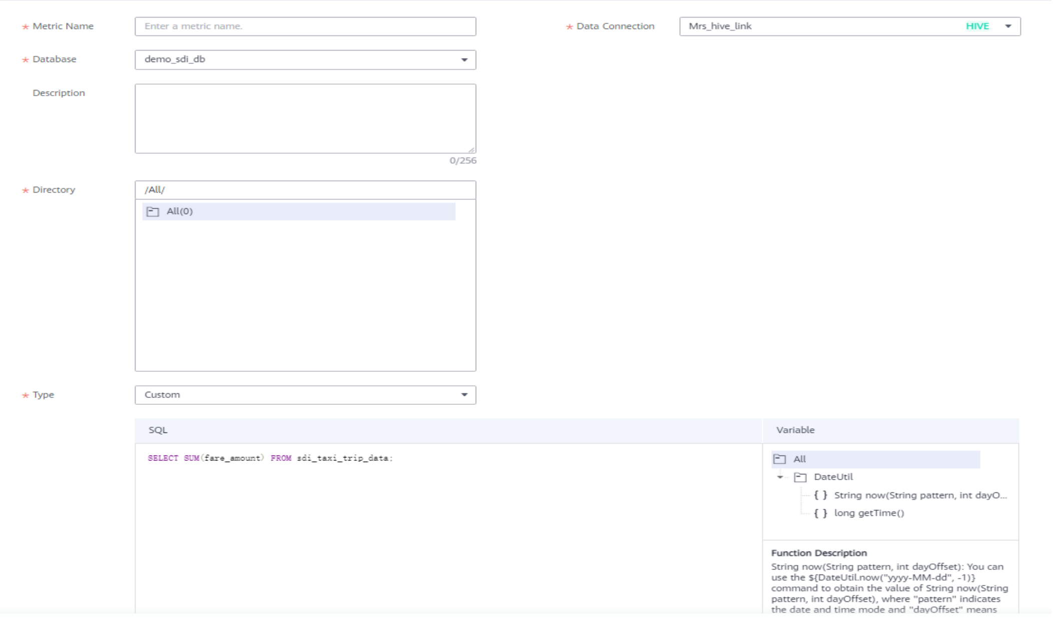
Step 2 Choose **Metric Monitoring** > **Metrics** from the left navigation bar.

Step 3 Click **Create**. In the displayed dialog box, set required parameters to create a metric.

The SQL statement is as follows:

```
SELECT SUM(fare_amount) FROM sdi_taxi_trip_data;
```

Figure 4-111 Creating a metric



* Metric Name: Enter a metric name.

* Data Connection: Mrs_hive_link HIVE

* Database: demo_sdi_db

Description: 0/256

* Directory: /All/
All(0)

* Type: Custom

SQL

```
SELECT SUM(fare_amount) FROM sdi_taxi_trip_data;
```

Variable

All
DateUtil
{ } String now(String pattern, int dayO...
{ } long getTime()

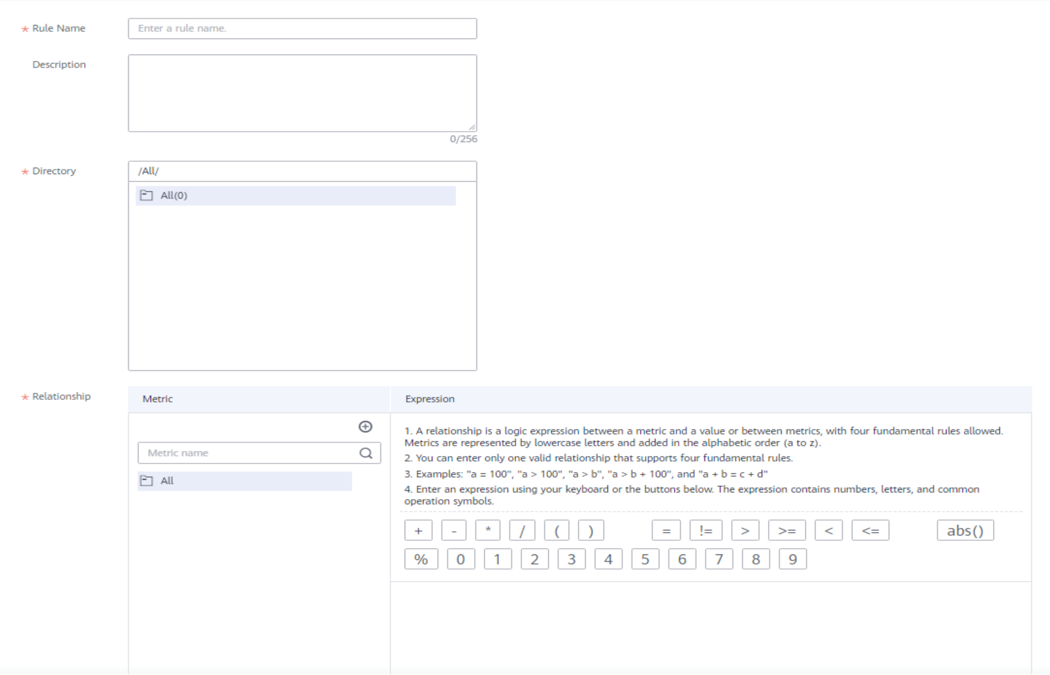
Function Description

String now(String pattern, int dayOffset): You can use the ${DateUtil.now("yyyy-MM-dd", -1)} command to obtain the value of String now(String pattern, int dayOffset), where "pattern" indicates the date and time mode and "dayOffset" means

Step 4 Choose **Metric Monitoring** > **Rule Management** from the left navigation bar.

Step 5 Click **Create**. In the displayed dialog box, set required parameters to create a rule.

Figure 4-112 Creating a rule



* Rule Name: Enter a rule name.

Description: 0/256

* Directory: /All/

All(0)

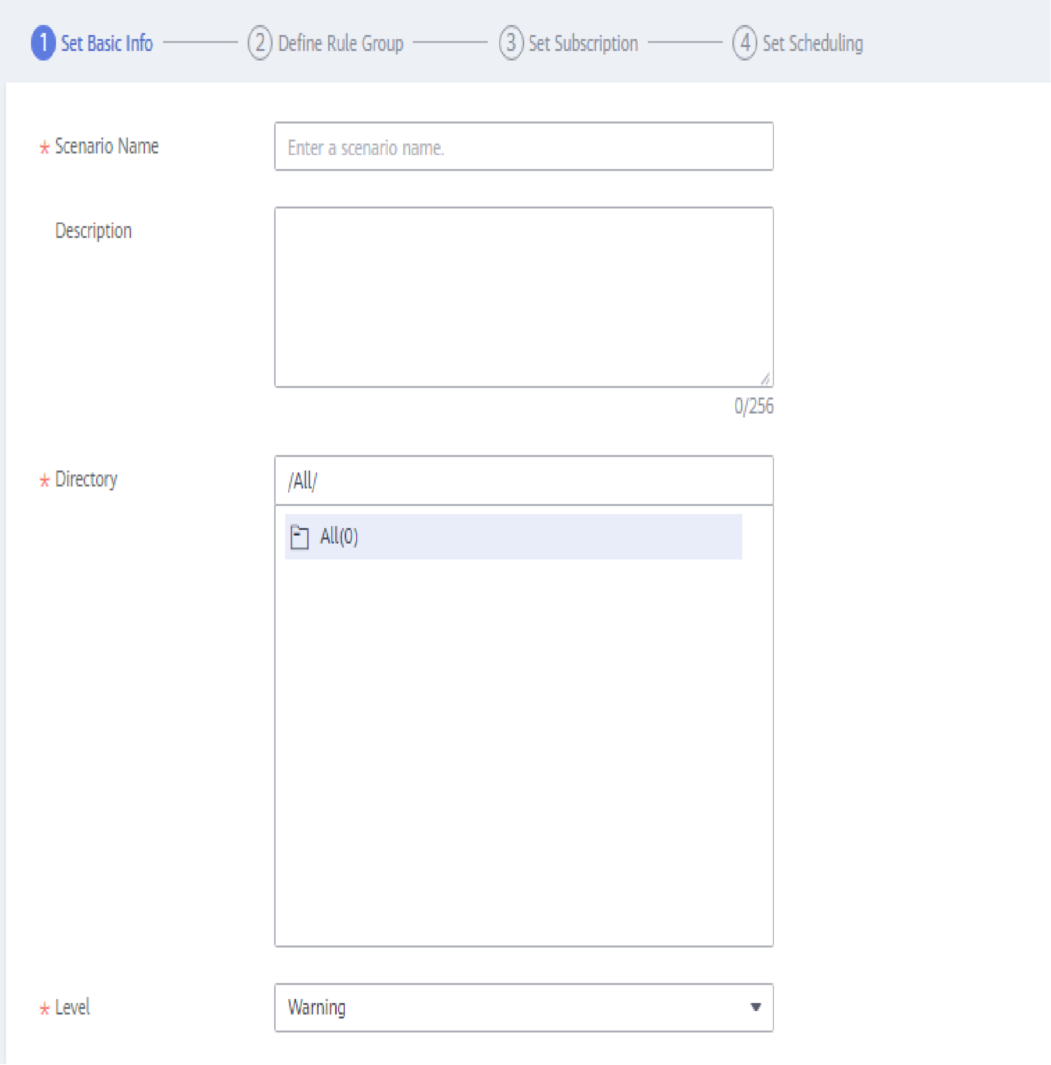
* Relationship

Metric	Expression
Metric name All	1. A relationship is a logic expression between a metric and a value or between metrics, with four fundamental rules allowed. Metrics are represented by lowercase letters and added in the alphabetic order (a to z). 2. You can enter only one valid relationship that supports four fundamental rules. 3. Examples: "a = 100", "a > 100", "a > b", "a > b + 100", and "a + b = c + d" 4. Enter an expression using your keyboard or the buttons below. The expression contains numbers, letters, and common operation symbols.

Step 6 Choose **Metric Monitoring** > **Business Scenario Management** from the left navigation bar.

Step 7 Click **Create**. In the displayed dialog box, set required parameters to create a scenario.

Figure 4-113 Setting basic information



1 Set Basic Info — 2 Define Rule Group — 3 Set Subscription — 4 Set Scheduling

* Scenario Name: Enter a scenario name.

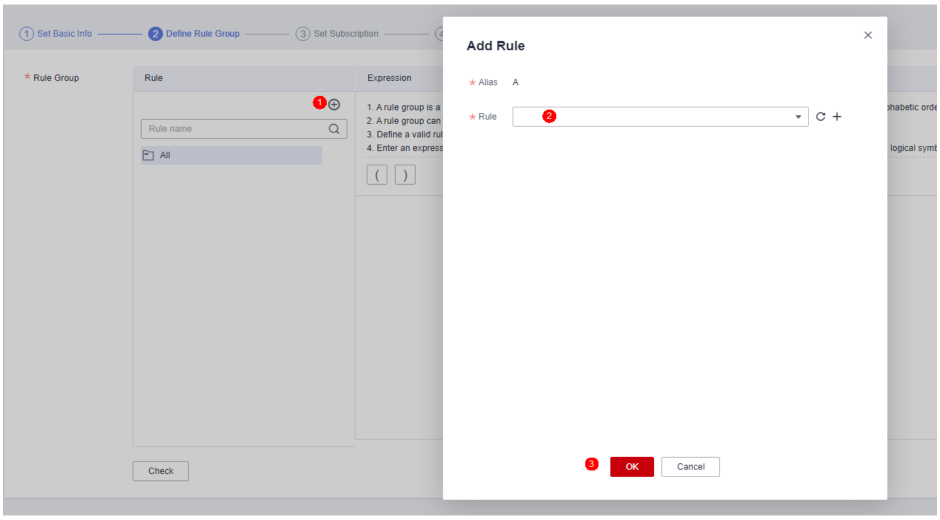
Description: 0/256

* Directory: /All/

All(0)

* Level: Warning

Figure 4-114 Defining a rule group



1 Set Basic Info — 2 Define Rule Group — 3 Set Subscription

* Rule Group

Rule	Expression
Rule name All	1. A rule group is a 2. A rule group can 3. Define a valid rul 4. Enter an express

Add Rule

* Alias: A

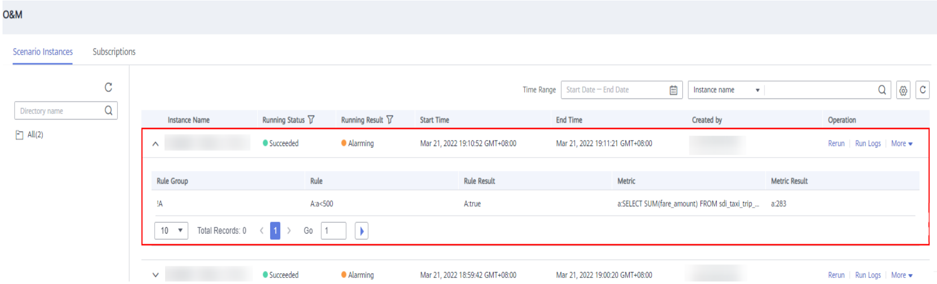
* Rule

Click **Next** and select **Once** or **On schedule** for **Repeat**.

Step 8 In the scenario list, locate the created scenario and click **Run** in the **Operation** column.

Step 9 Click the execution result to view the monitored metric.

Figure 4-115 Execution result



Instance Name	Running Status	Running Result	Start Time	End Time	Created by	Operation
[redacted]	● Succeeded	● Alarming	Mar 21, 2022 19:10:52 GMT+08:00	Mar 21, 2022 19:11:21 GMT+08:00	[redacted]	Rerun | Run Logs | More ▾
[redacted]	● Succeeded	● Alarming	Mar 21, 2022 18:59:42 GMT+08:00	Mar 21, 2022 19:00:20 GMT+08:00	[redacted]	Rerun | Run Logs | More ▾

Rule Group	Rule	Rule Result	Metric	Metric Result
!A	A:a<500	A:true	a:SELECT SUM(fare_amount) FROM sdi_taxi_trip_...	a:283

10 ▾ Total Records: 0 ‹ 1 › Go 1

📖 **NOTE**

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.

----End

4.9 Step 8: DataArts Catalog

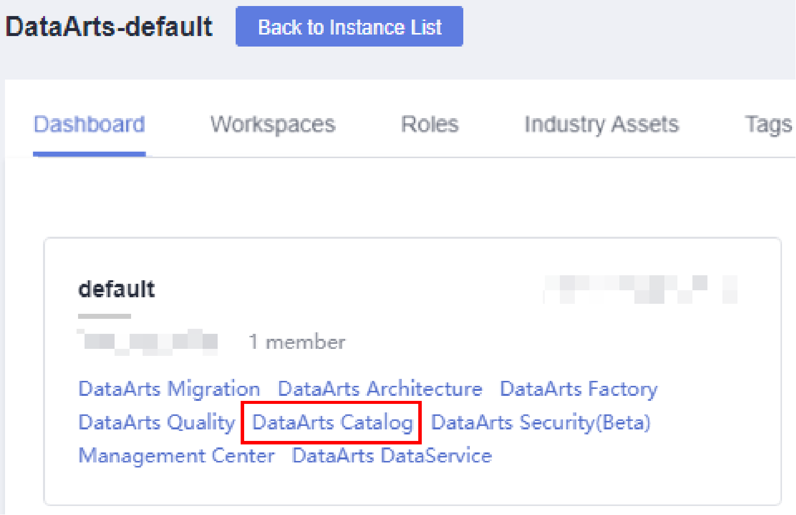
In the DataArts Studio DataArts Catalog module, you can view data maps. For details, see **DataArts Catalog**. A data map displays business assets and technical assets. Business assets refer to logical entities and business objects. Technical assets refer to data connections and database objects.

This topic describes how to view service assets and technical assets in the DataArts Catalog module of DataArts Studio. For example, in the fact table of a technical asset, you can view details such as data lineage. In the summary table of a technical asset, you can view details such as preview results.

Viewing Logical Assets and Technical Assets

Step 1 On the DataArts Studio console, locate an instance and click **Access**. On the displayed page, locate a workspace and click **DataArts Catalog**.

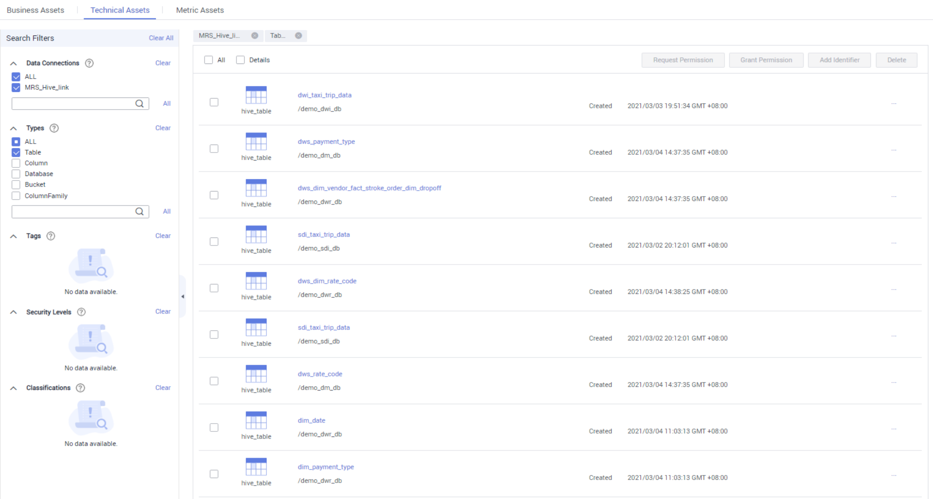
Figure 4-116 DataArts Catalog



Step 2 In the left navigation pane, choose **Data Map** > **Data Catalog**, click the **Logical Assets** tab, and select a business catalog under **Search Filters**. The logical assets that meet the filter criteria are displayed on the right.

Step 3 Click the **Technical Assets** tab, select a value for **Data Connections**, and select **Table** for **Types**. The metadata that meets the filter criteria is displayed on the right.

Figure 4-117 Technical assets



Data Connections
☑ ALL
☑ MRS_Hive_link

Types
■ ALL
☑ Table
□ Column
□ Database
□ Bucket
□ ColumnFamily

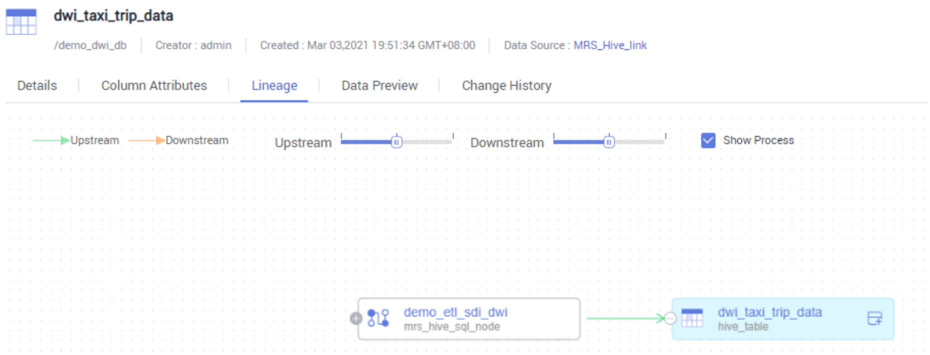
□ All □ Details

□	hive_table	dwi_taxi_trip_data /demo_dwi_db	Created	2021/03/03 19:51:34 GMT +08:00	...
□	hive_table	dws_payment_type /demo_dm_db	Created	2021/03/04 14:37:35 GMT +08:00	...
□	hive_table	dws_dim_vendor_fact_stroke_order_dim_dropoff /demo_dwr_db	Created	2021/03/04 14:37:35 GMT +08:00	...
□	hive_table	sdi_taxi_trip_data /demo_sdi_db	Created	2021/03/02 20:12:01 GMT +08:00	...
□	hive_table	dws_dim_rate_code /demo_dwr_db	Created	2021/03/04 14:38:25 GMT +08:00	...
□	hive_table	sdi_taxi_trip_data /demo_sdi_db	Created	2021/03/02 20:12:01 GMT +08:00	...
□	hive_table	dws_rate_code /demo_dm_db	Created	2021/03/04 14:37:35 GMT +08:00	...
□	hive_table	dim_date /demo_dwr_db	Created	2021/03/04 11:03:13 GMT +08:00	...
□	hive_table	dim_payment_type /demo_dwr_db	Created	2021/03/04 11:03:13 GMT +08:00	...

Step 4 In the asset list, click the name of the target metadata to view its details.

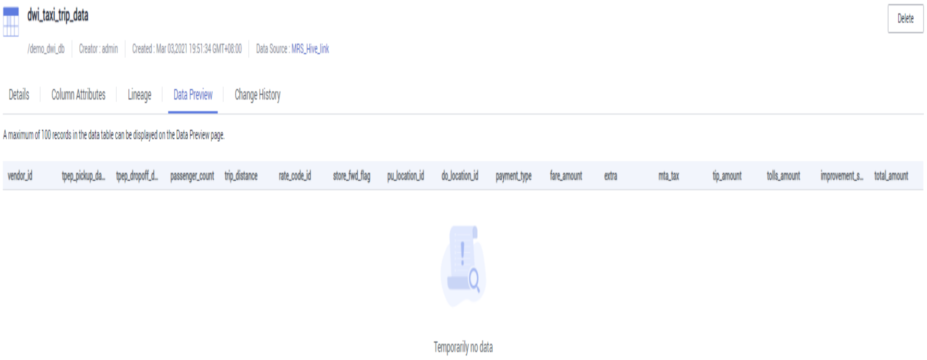
For example, click the name of the fact table **fact_stroke_order** in the asset list to view its details. On the details page, click the **Lineage** tab to view upstream and downstream information of the fact table.

Figure 4-118 Lineage



In the asset list, click the name of the summary table **dws_payment_type** to view its details. On the details page, click the **Data Preview** tab to preview the data in the summary table.

Figure 4-119 Data Preview



A maximum of 100 records in the data table can be displayed on the Data Preview page.

vendor_id	tpep_pickup_da...	tpep_dropoff_d...	passenger_count	trip_distance	rate_code_id	store_fwd_flag	pu_location_id	do_location_id	payment_type	fare_amount	extra	mta_tax	tip_amount	tolls_amount	improvement_s...	total_amount

Temporarily no data

----End

4.10 (Optional) Step 9: Service Unsubscription

In this development scenario, DataArts Studio, OBS, MRS, and GaussDB(DWS) incur fees. If you configure notifications, you may be billed for the following services:

- SMN: If you enable SMN notifications for your DataArts Studio modules, you need to pay for the notifications.
- EIP: If you use an EIP for your DataArts Migration cluster or DataArts DataService Exclusive cluster, you need to pay for the EIP.

After the development is complete, unsubscribe from DataArts Studio and other related services and delete resources in a timely manner to avoid undesired fees.

Table 4-22 Unsubscription methods for services

Service	Billing	Unsubscription Method
DataArts Studio	**DataArts Studio Billing**	DataArts Studio instances support only the yearly/monthly billing mode. You can unsubscribe from the package by referring to **Unsubscriptions**.
OBS	**OBS Billing**	OBS supports pay-per-use and yearly/monthly billing modes. Packages cannot be unsubscribed. In this example, the pay-per-use billing mode is used. You can delete the created bucket after using it. In addition, DataArts Studio job logs and DLI dirty data are stored in an OBS bucket named **dlf-log-{***Project id***}** by default. You can delete the bucket after unsubscribing from DataArts Studio.
MRS	**MRS Billing**	MRS supports pay-per-use and yearly/monthly billing modes. In this example, the pay-per-use billing mode is used. You can delete the MRS cluster after you finish with it. If you chose the yearly/monthly billing mode, you can unsubscribe from the yearly/monthly package you bought and delete the MRS cluster after you finish with it by referring to **Unsubscriptions**.
DWS	**GaussDB(DWS) Billing**	GaussDB(DWS) supports pay-per-use and yearly/monthly billing modes. In this example, the pay-per-use billing mode is used. You can delete the GaussDB(DWS) cluster after you finish with it. If you chose the yearly/monthly billing mode, you can unsubscribe from the yearly/monthly package you bought and delete the GaussDB(DWS) cluster after you finish with it by referring to **Unsubscriptions**.
SMN	**SMN Billing**	You pay only for what you use. After you unsubscribe from DataArts Studio, no notification will be generated. You can also delete the topics and subscriptions that have been generated.
EIP	**EIP Billing**	EIP supports the pay-per-use and yearly/monthly billing modes. In this example, the pay-per-use billing mode is used. You can release the EIP after you finish with it. If you chose the yearly/monthly billing mode, you can unsubscribe from the yearly/monthly package you bought and release the EIP after you finish with it by referring to **Unsubscriptions**.