

Data Lake Insight

User Guide

Issue 01
Date 2022-06-30



Copyright © Huawei Technologies Co., Ltd. 2022. All rights reserved.

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Technologies Co., Ltd.

Trademarks and Permissions



HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

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

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

Contents

1 Overview	1
2 Quick Start	3
2.1 Submitting a SQL Job	3
2.2 Submitting a Spark Job	5
2.3 Submitting a Flink SQL Job	7
3 DLI Console Overview	14
4 SQL Editor	16
5 Job Management	22
5.1 SQL Job Management	22
5.2 Flink Job Management	26
5.2.1 Flink Job Management	26
5.2.2 Managing Flink Job Permissions	29
5.2.3 Preparing Data	31
5.2.4 Creating a Flink SQL job	32
5.2.5 Creating a Custom Flink Job	37
5.2.6 Debugging a Job	40
5.2.7 Performing Operations on a Job	41
5.2.8 Job Details	45
5.3 Spark Job Management	48
5.3.1 Spark Job Management	48
5.3.2 Creating a Spark Job	49
6 Queue Management	54
6.1 Overview	54
6.2 Queue Permission Management	57
6.3 Creating a Queue	59
6.4 Deleting a Queue	61
6.5 Modifying the CIDR Block	62
6.6 Modifying Queue Specifications (Manual Scale-out/Scale-in)	62
6.7 Scheduling CU Changes (Periodic Scale-out/Scale-in)	64
6.8 Testing Address Connectivity	66
6.9 Creating a Message Notification Topic	67

7 Data Management.....	68
7.1 Databases and Tables.....	68
7.1.1 Overview.....	68
7.1.2 Database Permission Management.....	71
7.1.3 Table Permission Management.....	75
7.1.4 Creating a Database or a Table.....	82
7.1.5 Deleting a Database or a Table.....	89
7.1.6 Modifying the Owners of Databases and Tables.....	90
7.1.7 Importing Data to the Table.....	90
7.1.8 Exporting Data from DLI to OBS.....	93
7.1.9 Viewing Metadata.....	95
7.1.10 Previewing Data.....	96
7.2 Package Management.....	97
7.2.1 Overview.....	97
7.2.2 Managing Permissions on Packages and Package Groups.....	98
7.2.3 Creating a Package.....	100
7.2.4 Deleting a Package.....	101
7.2.5 Modifying the Owner.....	101
7.2.6 Built-in Dependency Packages.....	102
8 Job Templates.....	115
8.1 SQL Template Management.....	115
8.2 Flink Template Management.....	119
9 Datasource Connections.....	124
9.1 Datasource Connection and Cross-Source Analysis.....	124
9.2 Enhanced Datasource Connections.....	128
9.3 Managing Datasource Connection Permissions.....	135
9.4 Datasource Authentication.....	136
10 Global Configuration.....	141
10.1 Global Variables.....	141
10.2 Service Authorization.....	142
11 Using UDFs.....	144

1 Overview

What Is DLI?

Data Lake Insight (DLI) is a Serverless data processing and analysis service fully compatible with Apache Spark and Flink ecosystems. It frees you from managing any server. DLI supports standard SQL and is compatible with Spark and Flink SQL. It also supports multiple access modes, and is compatible with mainstream data formats. DLI supports SQL statements and Spark applications for heterogeneous data sources, including CloudTable, RDS, DWS, CSS, OBS, custom databases on ECSs, and offline databases.

Advantages

- **Full SQL compatibility**
You do not require a background in big data to do big data analysis. You only need to know SQL, and you are good to go. The SQL syntax is fully compatible with the standard ANSI SQL 2003.
- **Storage-compute decoupling**
DLI decouples storage from computing, and separates application and billing, so that you can use lower costs while improving resource utilization.
- **Enterprise multi-tenancy**
You can manage compute- or resource-related permissions by project or user, and implement fine-grained control to isolate data for each task.
- **O&M-free and high availability**
DLI frees you from the hassle of complicated O&M and upgrade operations while you enjoy high data availability with dual-AZ deployment.

DLI Core Engine: Spark+Flink

- **Spark** is a unified analysis engine that is ideal for large-scale data processing. It focuses on query, compute, and analysis. DLI optimizes performance and reconstructs services based on open-source Spark. It is compatible with the Apache Spark ecosystem and interfaces, and improves performance by 2.5x when compared with open-source Spark. In this way, DLI enables you to perform query and analysis of EB's of data within hours.
- **Flink** is a distributed compute engine that is ideal for batch processing, that is, for processing static data sets and historical data sets. You can also use it for

stream processing, that is, processing real-time data streams and generating data results in real time. DLI enhances features and security based on the open-source Flink and provides the Stream SQL feature required for data processing.

Serverless Architecture

DLI is a serverless big data query and analysis service. It has the following advantages:

- Pay-per-use: You pay only for what you use (scanned data volume/CUH packages). When no jobs are running, you will not be billed.
- Auto scaling: DLI ensures you always have enough capacity on hand to deal with any traffic spikes.

2 Quick Start

2.1 Submitting a SQL Job

Use DLI to submit a SQL job to query data. The general procedure is as follows:

Step 1: Logging in to the Cloud

Step 2: Uploading Data to OBS

Step 3: Logging In to the DLI Management Console

Step 4: Creating a Queue

Step 5: Creating a Database

Step 6: Creating a Table

Step 7: Querying Data

The following illustrates how to query OBS data using DLI. Operations of DLI data query are similar.

Step 1: Logging in to the Cloud

1. Open the DLI home page.
2. On the login page, enter the **Username** and **Password**, and click **Login**.

Step 2: Uploading Data to OBS

DLI allows you to query data stored on OBS. Before querying the data, you need to upload the data file to OBS.

1. In the services displayed, click **Object Storage Service** in **Storage**.
2. The OBS console page is displayed.
3. Create a bucket. The bucket name is globally unique. In this example, assume that the bucket name is **obs1**.
 - a. Click **Create Bucket** in the upper right corner.
 - b. On the **Create Bucket** page that is displayed, enter the **Bucket Name**.

- c. Click **Create Now**.
4. Click **obs1** to switch to the **Summary** page.
5. In the left navigation tree, click **Objects**. Click **Upload Object**. In the displayed dialog box, drag files or folders to upload or add file, for example, **sampledata.csv** to the file upload box. Then, click **Upload**.

After the file is uploaded successfully, the file path to be analyzed is **obs://obs1/sampledata.csv**.

For more information about OBS, see the *Object Storage Service Console Operation Guide*.


You are advised to use an OBS tool, such as OBS Browser+ or obsutil, to upload large files because OBS Console has restrictions on the file size and quantity. OBS Browser+ is a graphical tool that provides complete functions for managing your buckets and objects in OBS. You are advised to use this tool to create buckets or upload objects. obsutil is a command line tool for accessing and managing OBS resources. If you are familiar with command line interface (CLI), obsutil is recommended as an ideal tool for batch processing and automated tasks. For details about how to upload files to OBS, see the *OBS Tool Guide*.

Step 3: Logging In to the DLI Management Console

1. In the service list displayed, click **Data Lake Insight in Enterprise Intelligence**.
2. The DLI management console page is displayed. If you log in to the DLI management console for the first time, you need to be authorized to access OBS.

Step 4: Creating a Queue

A queue is the basis for using DLI. Before executing an SQL job, you need to create a queue.

- An available queue **default** is preset in DLI.
- You can also create queues as required.
 - a. On the DLI management console, click **SQL Editor** in the navigation pane on the left. The **SQL Editor** page is displayed.
 - b. On the left pane of the displayed **SQL Editor** page, click  to the right of **Queues**.

For details about how to create a queue, see [Creating a Queue](#).

Step 5: Creating a Database

Before querying data, create a database, for example, **db1**.

NOTE

The **default** database is a built-in database. You cannot create the database named **default**.

1. On the DLI management console, click **SQL Editor** in the navigation pane on the left. The **SQL Editor** page is displayed.

2. In the editing window on the right of the **SQL Editor** page, enter the following SQL statement and click **Execute**. Read and agree to the privacy agreement, and click **OK**.

```
create database db1;
```

After database **db1** is successfully created, **db1** will be displayed in the **Database** drop-down list box of **Databases**.

NOTE

When you execute a query on the DLI management console for the first time, you need to read the privacy agreement. You can perform operations only after you agree to the agreement. For later queries, you will not need to read the privacy agreement again.

Step 6: Creating a Table

After database **db1** is created, create a table (for example, **table1**) based on the sample data file **obs://obs1/sampleddata.csv** stored on OBS in **db1**.

1. In the upper right corner of the **SQL Editor** page, select the queue **default** and database **db1**.
2. Enter the following SQL statement in the job editor window and click **Execute**.

```
create table table1 (id int, name string) using csv options (path 'obs://obs1/sampleddata.csv');
```

After the table is created, click the **Databases** tab and then select **db1**. The created table **table1** is displayed in the lower part of the **Table** area.

Step 7: Querying Data

After performing the preceding steps, you can start querying data.

1. In the **Table** area on the **SQL Editor** page, double-click the created table **table1**. The SQL statement is automatically displayed in the SQL job editing window in the right pane. The following statement shows how to query 1,000 records in the **table1** table:

```
select * from db1.table1 limit 1000;
```

2. Click **Execute**. The system starts the query.

After the SQL statement is executed successfully, you can view the query result in **View Result** under the SQL job editing window.

2.2 Submitting a Spark Job

Use DLI to submit Spark jobs for real-time computing. The general procedure is as follows:

[Step 1: Logging in to the Cloud](#)

[Step 2: Uploading Data to OBS](#)

[Step 3: Logging In to the DLI Management Console](#)

[Step 4: Creating a Queue](#)

[Step 5: Creating a Package](#)

[Step 6: Submitting a Spark Job](#)

Step 1: Logging in to the Cloud

To use DLI, you need to log in to the cloud.

1. Open the DLI home page.
2. On the login page, enter the **Username** and **Password**, and click **Login**.

Step 2: Uploading Data to OBS

Before submitting Spark jobs, upload data files to OBS.

1. In the services displayed, click **Object Storage Service** in **Storage**.
2. The OBS console page is displayed.
3. Create a bucket. The bucket name is globally unique. In this example, assume that the bucket name is **obs1**.
 - a. Click **Create Bucket**.
 - b. On the **Create Bucket** page that is displayed, specify **Bucket Name**.
 - c. Click **Create Now**.
4. Click **obs1** to switch to the **Summary** page.
5. From the left navigation tree, click **Object**. Click **Upload Object**. In the displayed dialog box, drag files or folders to upload or add file to the file upload box, for example, **spark-examples.jar**. Then, click **Upload**.

After the file is uploaded successfully, the file path to be analyzed is **obs://obs1/spark-examples.jar**.

For more information about OBS, see the *Object Storage Service Console Operation Guide*.

You are advised to use an OBS tool, such as OBS Browser+ or obsutil, to upload large files because OBS Console has restrictions on the file size and quantity. OBS Browser+ is a graphical tool that provides complete functions for managing your buckets and objects in OBS. You are advised to use this tool to create buckets or upload objects. obsutil is a command line tool for accessing and managing OBS resources. If you are familiar with command line interface (CLI), obsutil is recommended as an ideal tool for batch processing and automated tasks. For details about how to upload files to OBS, see the *OBS Tool Guide*.

Step 3: Logging In to the DLI Management Console

To submit Spark jobs, you need to enter the Spark job creation page first.

1. In the service list displayed, click **Data Lake Insight** in **Enterprise Intelligence**.
2. The DLI management console page is displayed. If you log in to the DLI management console for the first time, you need to be authorized to access OBS.

Step 4: Creating a Queue

If it is your first time submitting a Spark job, create a queue first. For example, create a queue named **test**. For details about how to create a queue, see [Creating a Queue](#).

Step 5: Creating a Package

Before submitting a Spark job, you need to create a package, for example, **spark-examples.jar**. For details, see [Creating a Package](#).

Step 6: Submitting a Spark Job

1. On the DLI management console, choose **Job Management > Spark Jobs** in the navigation pane on the left. The page for creating a Spark job is displayed.
2. On the Spark job editing page, set related parameters. For details, see [GUI Description](#).
3. Click **Execute** in the upper right corner of the Spark job editing window, read and agree to the privacy agreement, and click **OK**. Submit the job. A message is displayed, indicating that the job is successfully submitted.
4. (Optional) Switch to the **Job Management > Spark Jobs** page to view the status and logs of the submitted Spark job.

NOTE

When you click **Execute** on the DLI management console for the first time, you need to read the privacy agreement. Once agreed to the agreement, you will not receive any privacy agreement messages for subsequent operations.

2.3 Submitting a Flink SQL Job

Use DLI to submit Flink SQL jobs for real-time computing. The general procedure is as follows:

[Step 1: Logging in to the Cloud](#)

[Step 2: Prepare Data Sources and Data Output Channels](#)

[Step 3: Create an OBS Bucket for Saving Outputs](#)

[Step 4: Logging In to the DLI Management Console](#)

[Step 5: Creating a Queue](#)

[Step 6: Creating an Enhanced Datasource Connection](#)

[Step 7: Creating a Datasource Authentication](#)

[Step 8: Creating a Flink SQL Job](#)

In the sample scenario, you need to create a Flink SQL job, and the job has an input and an output stream. The input stream is used to read data from DIS, and the output stream is used to write data to Kafka.

Step 1: Logging in to the Cloud

To use DLI, you need to log in to the cloud.

1. Open the DLI home page.
2. On the login page, enter the **Username** and **Password**, and click **Login**.

Step 2: Prepare Data Sources and Data Output Channels

DLI Flink jobs support other services as data sources and data output channels. For details, see [Preparing Data](#).

In this example, assume that the job name is JobSample and DIS is used as the data source. To enable DIS, see **Creating a DIS Stream** in the *Data Ingestion Service User Guide*. Use Kafka as the data output channel to create a Kafka platinum instance. For details, see **Buying an Instance** in the *Distributed Message Service for Kafka User Guide*.

- Create a DIS stream for the job input stream.
 - a. Log in to the DIS console.
 - b. In the upper left corner of the management console, select the target region and project.
 - c. On the homepage, click **Buy Stream** and configure stream parameters. The channel information is as follows:
 - **Region:** Select the region where DLI is located.
 - **Stream Name:** csinput
 - **Stream Type:** Common
 - **Partitions:** 1
 - **Data Retention (hours):** 24
 - **Source Data Type:** BLOB
 - **Auto Scaling:** disabled
 - **Advanced Settings:** Skip it.
 - d. Click **Buy Now**. The **Details** page is displayed.
 - e. Click **Submit**.
- Create a Kafka platinum instance for the job output stream.
 - a. Before creating a Kafka instance, you need to ensure the availability of resources, including a virtual private cloud (VPC), subnet, security group, and security group rules.
 - For details about how to create a VPC and subnet, see **Creating a VPC** in the *Virtual Private Cloud User Guide*. If you need to create and use a subnet in an existing VPC, see **Creating a VPC** in the *Virtual Private Cloud User Guide*.

NOTE

- The created VPC and the Kafka instance must be in the same region.
- Retain the default settings unless otherwise specified.
- For details about how to create a security group, see **Creating a Security Group** in the *Virtual Private Cloud User Guide*. For details about how to add rules to a security group, see **Creating a Security Group** in the *Virtual Private Cloud User Guide*.

For more information, see **Preparing Instance Dependency Resources** in the *Distributed Message Service for Kafka User Guide*.

- b. Log in to the Kafka management console.
- c. Select a region in the upper left corner.
- d. On the **Kafka Premium** page, click **Buy Kafka Instance** in the upper right corner and set related parameters. The instance information is as follows:
 - **Region:** Select the region where DLI is located.
 - **AZ:** Keep the default value.
 - **Instance Name:** kafka-dliflink
 - **Version:** Keep the default value.
 - **CPU Architecture:** Keep the default value.
 - **Bandwidth:** Keep the default value.
 - **Maximum Partitions:** Keep the default value.
 - **Storage Space:** Keep the default value.
 - **VPC:** vpc-dli
 - **Subnet:** dli-subnet
 - **Security Group:** Keep the default value.
 - **Capacity Threshold Policy:** Keep the default value.
 - **Manager Username:** dliflink (used to log in to the instance management page)
 - **Password:** **** (The system cannot detect your password.)
 - **Confirm Password:** ****
 - **More Settings:** Do not configure this parameter.
- e. Click **Buy Now**. The **Details** page is displayed.
- f. Click **Submit**.

Step 3: Create an OBS Bucket for Saving Outputs

In this example, you need to enable OBS for job JobSample to provide DLI Flink jobs with the functions of checkpoint, saving job logs, and commissioning test data.

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

1. In the navigation pane on the OBS management console, select **Object Storage**.

2. In the upper right corner of the page, click **Create Bucket** and set bucket parameters.
 - **Region:** Select the region where DLI is located.
 - **Data Redundancy Policy:** Single-AZ storage
 - **Bucket Name:** **smoke-test**
 - **Storage Class:** Standard
 - **Bucket Policy:** Private
 - **Default Encryption:** Disable
 - **Direct Reading:** Disable
 - **Tags:** Leave it blank.
3. Click **Create Now**.

Step 4: Logging In to the DLI Management Console

1. In the service list displayed, click **Data Lake Insight in Enterprise Intelligence**.
2. The DLI management console page is displayed. If you log in to the DLI management console for the first time, you need to be authorized to access OBS.

Step 5: Creating a Queue

When creating a DLI Flink SQL job, you cannot use the existing default queue. Instead, you need to create a queue, for example, a queue named **Flinktest**. For details about how to create a queue, see [Creating a Queue](#).

1. On the **Dashboard** page of the DLI management console, click **Buy Queue** in the upper right corner.
2. Configure parameters.
 - **Billing Mode:** Pay-per-use
 - **Region:** default region
 - **Name:** Flinktest
 - **Queue Usage:** For general purpose Select **Dedicated Resource Mode**.
 - **CPU Architecture:** x86
 - **CU Specifications:** 16 CUs
 - **Advanced Configuration:** Custom
 - **CIDR Block:** The configured CIDR block cannot conflict with the Kafka subnet CIDR block.
3. Click **Buy** to confirm the configuration.
4. Confirm the configuration and submit the request.

Step 6: Creating an Enhanced Datasource Connection

To create a DLI Flink job, you need to create an enhanced datasource connection. For details, see [Creating a Connection](#).

 NOTE

- The CIDR block of the DLI queue which is bound with a datasource connection cannot overlap with that of the data source.
 - Datasource connections cannot be created for the default queue.
 - To access a datasource connection table, you need to use the queue for which a datasource connection has been created.
1. In the navigation pane of the DLI management console, choose **Datasource Connections**.
 2. Click the **Enhanced** tab and click **Create** in the upper left corner. Set the following parameters:
 - **Connection Name:** diskafka
 - **Bind Queue:** Flinktest
 - **VPC:** vpc-dli
 - **Subnet:** dli-subnet
 3. Click **OK**.
 4. On the **Enhanced** tab page, click the created connection **diskafka** to view its ID and status. If the connection status is **Active**, the connection is successful.

Step 7: Creating a Datasource Authentication

To create a Kafka cluster, you need to enable SSL access. Download the authentication certificate and upload it to the customized OBS bucket. For details about how to create datasource authentication, see [Datasource Authentication](#).

1. On the DLI management console, click **Datasource Connections**.
2. On the **Datasource Authentication** tab page, click **Create** to create an authentication information. Set the following parameters:
 - **Authentication Certificate:** Flink
 - **Type:** Kafka_SSL
 - **Truststore Path:** obs://smoke-test/client.truststore.jks.
 - **Truststore Password:** *****

You do not need to set other parameters.

3. Click **OK**.

Step 8: Creating a Flink SQL Job

After the data source and data output channel are prepared, you can create a Flink SQL job.

1. In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.
2. In the upper right corner of the **Flink Jobs** page, click **Create Job**. Set the following parameters:
 - **Type:** Flink SQL
 - **Name:** DIS-Flink-Kafka
 - **Editor:** SQL editor

3. Click **OK** to enter the **Edit** page.
4. Edit a Flink SQL job

Enter details SQL statements in the SQL statement edit area. See the following:

```
CREATE sink STREAM car_info (  
  a1 string,  
  a2 string,  
  a3 string,  
  a4 INT  
)  
WITH (  
  type = "dis",  
  region = "",  
  channel = "csinput",  
  encode = "csv",  
  FIELD_DELIMITER = ";"  
);  
  
insert into car_info select 'id','owner','brand',1;  
insert into car_info select 'id','owner','brand',2;  
insert into car_info select 'id','owner','brand',3;  
insert into car_info select 'id','owner','brand',4;  
insert into car_info select 'id','owner','brand',5;  
insert into car_info select 'id','owner','brand',6;  
insert into car_info select 'id','owner','brand',7;  
insert into car_info select 'id','owner','brand',8;  
insert into car_info select 'id','owner','brand',9;  
insert into car_info select 'id','owner','brand',10;  
  
CREATE SOURCE STREAM car_info (  
  a1 string,  
  a2 string,  
  a3 string,  
  a4 INT  
)  
WITH (  
  type = "dis",  
  region = "",  
  channel = "csinput",  
  encode = "csv",  
  FIELD_DELIMITER = ";"  
);  
  
CREATE SINK STREAM kafka_sink (  
  a1 string,  
  a2 string,  
  a3 string,  
  a4 INT  
) // Output Field  
WITH (  
  type="kafka",  
  kafka_bootstrap_servers = "192.x.x.x:9093, 192.x.x.x:9093, 192.x.x.x:9093",  
  kafka_topic = "testflink", // Written topic  
  encode = "csv", // Encoding format, which can be JSON or CSV.  
  kafka_certificate_name = "Flink",  
  kafka_properties_delimiter = ",",  
  kafka_properties = "sasL.jaas.config=org.apache.kafka.common.security.plain.PlainLoginModule  
required username=\\"xxx\\" password=\\"xxx\\";sasL.mechanism=PLAIN;security.protocol=SASL_SSL"  
);  
  
INSERT INTO kafka_sink  
SELECT * FROM car_info;
```


5. Click **Check Semantics**.
6. Set job running parameters. Set the following parameters:

- **CUs:** 2
- **Job Manager CUs:** 1
- **Max Concurrent Jobs:** 1
- **Dirty Data Dump Address:** obs://smoke-test
- **Queue:** Flinktest

You do not need to set other parameters.

7. Click **Save**.
8. Click **Start**. On the displayed **Start Flink Job** page, confirm the job specifications and the price, and click **Start Now** to start the job.

After the job is started, the system automatically switches to the **Flink Jobs** page, and the created job is displayed in the job list. You can view the job status in the **Status** column. After a job is successfully submitted, **Status** of the job will change from **Submitting** to **Running**. After the execution is complete, the message **Completed** is displayed.

If **Status** of a job is **Submission failed** or **Running exception**, the job fails to be submitted or fails to run. In this case, you can move the cursor over the status icon in the **Status** column of the job list to view the error details. You can click  to copy these details. After handling the fault based on the provided information, resubmit the job.

9. After the job is complete, you can log in to the management console of Distributed Message Service for Kafka to view the corresponding Kafka premium instance. Click the instance name and click the **Message Query** tab to view the message body.

3 DLI Console Overview

Overview

Log in to DLI and go to the **Overview** page.

The following describes the functions of the **Overview** page, and the order follows the numeric values marked on the image.

Table 3-1 Areas

No.	Area	Description
1	Navigation pane	The navigation bar on the DLI management console, including Overview , SQL Editor , Job Management , Queue Management , Data Management , Job Templates , Datasource Connection , and Global Variables .
2	DLI introduction	Data Lake Insight (DLI) is a fully hosted big data processing and analysis service. It frees you from managing any server. DLI supports standard SQL and is compatible with Spark and Flink SQL. It also supports multiple access modes, and is compatible with mainstream data formats.
3	Create Queue	Click Create Queue to create a queue.
4	SQL job introduction	Click Create Job in this area or the upper right corner to access the SQL Editor page of the SQL job. For details about how to create a package, see SQL Editor .
5	Flink job introduction	Click Create Job in this area or in the upper right corner to go to the Flink Jobs page of the Flink jobs. For details about how to create a package, see Flink Job Management .
6	Spark job introduction	Click Create Job in this area or the upper right corner to access the Spark job creation page. For details, see Creating a Spark Job .

No.	Area	Description
8	Usage duration for all queues	This area displays the usage of all queues of the last day, week, or month.
9	Usage duration for one queue	This area displays the usage of a queue in the last week or month.
10	Helpful links	Including: <ul style="list-style-type: none">• <i>User Guide</i>• <i>API Reference</i>• <i>SDK Download</i>
11	Procedure of creating a job	<ol style="list-style-type: none">1. Creating a queue2. Preparing data3. Editing and submitting a job4. Viewing job status

4 SQL Editor

You can use SQL statements in the SQL job editor to execute data query. DLI supports SQL 2003 and complies with Spark SQL.

On the **Overview** page, click **SQL Editor** in the navigation pane on the left or **Create Job** in the upper right corner of the SQL job introduction. The **SQL Editor** page is displayed.

Enter the **SQL Editor**. A message is displayed, indicating that a temporary DLI data bucket will be created. The created bucket is used to store temporary data generated by DLI, such as job logs and job results. You cannot view job logs if you choose not to create it. You can periodically delete objects in a bucket or transit objects between different storage classes. The bucket name is set by default.

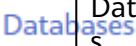
If you do not need to create a DLI temporary data bucket and do not want to receive this message, select **Do not show again** and click **Cancel**.






The areas and buttons on the **SQL Editor** page are described as follows:

Navigation pane

The navigation pane on the left contains **Databases** and **Queues**.

Table 4-1 Navigation pane button description

N o.	Tab/ Button	Tab/ Button Name	Description
1		Databases	Displays all the existing databases and tables in these databases. <ul style="list-style-type: none">• After you click a database name, the tables in the database are displayed.• After you click a table name, the metadata in the table is displayed below. A maximum of 20 metadata records can be displayed.• After you double-click a table name, the SQL query statement is automatically entered in the job editing window.

No.	Tab/ Button	Tab/ Button Name	Description
2		Queues	Displays existing queues.
3		Templates	Click the drop-down button to view the built-in SQL sample templates, including 22 standard TPC-H query statements.
4		Create	You can click the button on the related page to create a queue, database, or table. For details, see Creating a Queue and Creating a Database or a Table .
5		Refresh	Refreshes the existing queues, databases, and table lists.
6		Search	You can enter a keyword to search for the corresponding database and table.

SQL Job Editing Window

The SQL job editing window is displayed in the upper right corner. The following describes the functions of buttons in the SQL editing window.

Table 4-2 Description of buttons in the SQL job editing window

No.	Button	Description
2	Queue	Select a queue from the drop-down list box. If no queue is available, the message Select a queue is displayed. Create a queue first.
3	Database	Select a database from the drop-down list box. If no database is available, the message "Please select a database" is displayed. Create a database first. NOTE If the database where the table is located has been specified in your entered SQL statement, the selected database will become invalid.
4	Execute	You can click this button to run the SQL statements in the job editing window.
5	Format	Format SQL statements.
6	Syntax Reference	Click the button to switch to the <i>Data Lake Insight SQL Syntax Reference</i> .
7	Settings	Set the properties for submitted SQL jobs in the key/value format. A maximum of 10 attributes can be set.

No.	Button	Description
8	More	Including: <ul style="list-style-type: none">• Click Verify Syntax to check whether the SQL statement is correct.• Click Set as Template to set a common SQL statement as a template. For details, see SQL Template Management.• Click Choose Template to select the SQL statement that has been saved as a template.• Click Change Theme to select the black or white background.
9	SQL statement editing area	Area for you to write and edit SQL statements.
10	Cursor position	The line and column where the cursor is in the job editing area.
11	Keyboard shortcuts	For details about keyboard shortcuts, see Table 4-3 .

Table 4-3 Keyboard shortcut description

Keyboard Shortcut	Description
Ctrl+R or Ctrl+Enter	Execute SQL statements. You can run SQL statements by pressing Ctrl+R or Ctrl + Enter on the keyboard.
Ctrl+F	Format SQL statements. You can press Ctrl + F to format the SQL statement.
Ctrl+Q	Syntax verification. You can press Ctrl + Q to verify the syntax of SQL statements.
Ctrl+Z	Undo your operation. You can press Ctrl + Z to undo the SQL statement in the job editing window to the previous step. For example, undo the formatted SQL statement to the state before formatting. This function is similar to the undoing operation.

Keyboard Shortcut	Description
Tab	Autocomplete. You can press Tab to quickly find the required SQL statement keywords, database name, and table name. <ul style="list-style-type: none">• Before pressing the Tab key, ensure that the cursor is located in the job editing window.• If there are no statements in the job editing window, the prompted information is sequenced in alphabetical order.• If you press Tab after entering the first letter of a keyword in the job editing window, the prompted information contains the SQL statement keywords, database names, and table names, whose first letter is the entered letter.
F11	Full screen. You can press F11 to display the SQL Job Editor window in full screen. Press F11 again to leave the full screen.

Execution Queries and View Result

The **Executed Queries (Last 7 Days)** and **View Result** tab pages are displayed in the lower right corner of the page.

- Executed Queries

Table 4-4 Area description

Area	Description
Executed Queries (Last 7 Days)	Executed Queries (Last 7 Days): Displays information about jobs submitted in the last 7 days. Including: <ul style="list-style-type: none">• Name• Created• Status• Statement• Operation<ul style="list-style-type: none">- Modify: Re-edit the SQL statement.- Cancel: Stop executing the SQL statement.- SparkUI: Switch to the SparkUI page to view the SQL statement execution process.- Export Log: Export SQL statement execution logs.

You can filter the execution history in the following ways:

- In the upper right corner, select a queue name or enter an execution statement.
 - In the list, click the icon next to **Created** and choose **Ascending** or **Descending**.
 - Select a job status from the list.
- View Result

Table 4-5 Area description

No	Button	Description
1	Clear All	Clear the displayed SQL statement query result.
2	Graphical Result/Table Result	Display the query result in a graph or table.
3	Export Result	Exports the query result to OBS. For details, see Exporting Query Results .

SQL Query Procedure

1. Log in to the Data Lake Insight management console, select **SQL Jobs**, and click **Create Job** to go to the **SQL Editor** page.
2. After you enter the **SQL Editor** page, the system prompts you to create a temporary DLI data bucket to store temporary data generated by DLI, such as job logs and job results. You cannot view job logs if you choose not to create it. Click **OK**. The system displays a message indicating that the temporary bucket is created successfully.
3. Select the queue to be used from the queue list in the upper left corner of the SQL job editing window. The **default** queue is used. For details about how to create a queue, see [Creating a Queue](#).
4. In the upper right corner of the SQL job editing window, select a database, for example, **qw**, from the **Databases** drop-down list.
5. Create a table, for example, **qw**. For details about how to create a database and table, see [Creating a Database or a Table](#).
6. In the SQL job editing window, enter the following SQL statement:

```
SELECT * FROM qw.qw LIMIT 10;
```

Alternatively, you can double-click the table name **qw** on the left. The query statement is automatically entered in the SQL job editing window.
7. Click **Verify Syntax** in **More** to check whether the SQL statement is correct.
 - a. If the verification fails, check the SQL statement accuracy by referring to *Data Lake Insight SQL Syntax Reference*.
 - b. If the syntax verification is successful, click **Execute**. Read and agree to the privacy agreement. Click **OK** to execute the SQL statement.
 - c. After the SQL statement is executed successfully, you can view the execution result in the area under the SQL job editing window.

8. (Optional) In the **View Result** tab page, click **Graphical Result** in the upper right corner to display the result in a graph. Then click **Table Result** to switch to the table form.

 **NOTE**

- If no column of the numeric type is displayed in the execution result, the result cannot be represented in charts.
- The chart types include the bar chart, line chart, and fan chart.
- In the bar chart and line chart, the X axis can be any column, while the Y axis supports only columns of the numeric type. The fan chart displays the corresponding legends and indicators.

Operation Skills in the Job Editing Window

- A simple method for importing SQL statements in the SQL job editing window:
 - Double-click a table name in the navigation pane on the left to import the query statement of the selected table into the SQL statement editing window, and then click **Execute** to query.
 - You can drag table names and column names to the job editing window to compile SQL statements.
 - You can click **More** and choose **Save as Template** to save the SQL statement as a template for future use.

To use the SQL statement, click **More > Choose Template**. Double-click the required SQL statement in the existing templates, import it to the SQL statement editing window, and click **Execute**, or modify it as required before executing the SQL statement.

- SQL statements can be executed in batches.
- Features of the job editing window
The job editing window has the following three features:
 - **Highlighting:** Common syntax is highlighted in different colors.
 - **Comment support:** You can write comments in a single line or multiple lines. In each line, the comments follow two consecutive hyphens (--).
 - **Cursor location display:** After you click in a line in the job editing window, the location of your cursor in the job editing window, including the column number and line number, will be automatically displayed in the lower right corner of the window.

5 Job Management

5.1 SQL Job Management

SQL jobs allow you to execute SQL statements entered in the SQL job editing window, import data, and export data.

SQL job management provides the following functions:

- [Searching for a Job](#)
- [Viewing Job Details](#)
- [Terminating a Job](#)
- [Exporting Query Results](#)


In addition, you can click **Quick Links** to switch to the details on *User Guide*.

Job Management page

On the **Overview** page, click **SQL Jobs** to go to the SQL job management page. Alternatively, you can click **Job Management > SQL Jobs**. The job list displays all SQL jobs. If there are a large number of jobs, they will be displayed on multiple pages. You can switch to the specified page as required. DLI allows you to view jobs in all statuses. By default, jobs in the job list are displayed in descending order according to the job creation time, with the most recently created jobs displayed at the top.

Table 5-1 Job management parameters

Parameter	Description
Queue	Name of the queue to which a job belongs
Created	Time when a job is created. Jobs can be displayed in ascending or descending order of the job creation time.

Parameter	Description
Job Type	Job type. The following values are available: <ul style="list-style-type: none">● IMPORT: Indicates a job used for importing data to DLI.● EXPORT: Indicates a job used for exporting data from DLI.● DCL: Involves conventional DCLs and operations related to queue permissions.● DDL: Is consistent with conventional DDL operations, including creating and deleting databases and tables.● QUERY: Indicates a job used for querying data by running SQL statements.● INSERT: Indicates a job used for inserting data by running SQL statements.● UPDATE: Updates data.● DELETE: Deletes a SQL job.● DATA_MIGRATION: Indicates cloud data migration.● RESTART_QUEUE: Restarts a queue.● SCALE_QUEUE: Indicates that the queue specifications are changed (capacity expansion or reduction).
Status	Job status. The following values are available: <ul style="list-style-type: none">● Running● Scaling● Submitting● Canceled● Finished● Failed
Statement	Displays the SQL statements for a job as well as descriptions about export and table creation operations. You can click  to copy the corresponding statement.
Duration	Running duration of a job


Parameter	Description
Operation	<ul style="list-style-type: none">● Cancel<ul style="list-style-type: none">– You can terminate a job only when the job is in Submitting or Running status.– A job whose status is Finished, Failed, or Canceled cannot be terminated.– If the Cancel button is gray, you are not allowed to perform this operation.● SparkUI: After you click this button, the Spark job execution page is displayed.● In addition to the preceding operations, the following operation is available for QUERY jobs and asynchronous DDL jobs.<ul style="list-style-type: none">– View Result: View the job running result.– Export Result: Export the job running result to the created OBS bucket. For details, see section Exporting Query Results.● In addition to the preceding operations, the EXPORT job also includes the following operations:<ul style="list-style-type: none">– Download● Archive Log: Save job logs to the temporary OBS bucket created by DLI.● Export Log: Export logs to the created OBS bucket. If the job is in the Running state, logs cannot be exported. <p>NOTE You have the permission to create OBS buckets.</p>

Searching for a Job

On the **SQL Jobs** page, you can filter jobs by using the following methods.

- Select a queue name.
- Set the date range.
- Enter the statement or job ID.
- Select the creation time in ascending or descending order.
- Select a job type.
- Select a job status
- Select the job execution duration in ascending or descending order.

Viewing Job Details

On the **SQL Jobs** page, you can click  in front of a job record to view details about the job.

Job details vary with job types. The following describes how to load data, create table, and select job.

- **load data** (job type: IMPORT) includes the following information: Queue name, job ID, creation time, job type, job status, execution statement, running duration, scanned data, execution user, number of results, number of scanned data records, number of error records, database name, table name, file format, table header, and quoted characters, separator, data source path, escape character, import start time, import end time, date format, and timestamp format.
- **create table** (job type: DDL) includes the following information: Queue name, job ID, creation time, job type, job status, execution statement, running duration, scanned data, and execution user.
- **select** (job type: QUERY) includes the following information: Queue name, job ID, creation time, job type, job status, execution statement, running duration, number of results (successful and exported results), scanned data, execution user, and result status (running successfully, viewing results, running failure, and failure cause).

Terminating a Job

On the **SQL Jobs** page, you can click **Terminate** in the **Operation** column to stop a submitting or running job.

Exporting Query Results

You can export results on the **SQL Jobs** page or the **SQL Editor** page.

- On the **Job Management > SQL Jobs** page, you can click **More < Export Result** in the **Operation** column to export the query result.
- After the query statement is successfully executed on the **SQL Editor** page, click **Export Result** on the right of the **View Result** tab page to export the query result.

NOTE

If no column of the numeric type is displayed in the query result, the result cannot be exported.

Table 5-2 Parameters

Parameter	Description
File Format	Format of the exported query result file. Formats other than JSON will be supported in later versions.
Queue	Select a queue.

Parameter	Description
Compression Format	Compression format of the data to be exported. The following options are supported: <ul style="list-style-type: none">• none• bzip2• deflate• gzip
Storage Path	Enter or select an OBS path to store the result. NOTE <ul style="list-style-type: none">• After selecting an OBS bucket, enter a name for the folder. If the folder does not exist, it will be created in OBS.• The folder name cannot contain the special characters of \ / : * ? "< > , and cannot start or end with a dot (.).
Export Mode	Mode for saving the exported query result data. <ul style="list-style-type: none">• New OBS directory: If the specified export directory exists, an error is reported and the export operation cannot be performed.• Existing OBS directory (Overwritten): If you create a file in the specified directory, the existing file will be overwritten.
Number of Results	Number of exported query results. If no value is entered or the value is 0, all results are exported.
Advanced Settings	Table Header: No/Yes Whether the data to be exported contains table headers. This parameter is invalid only when Export Format is set to json . Formats other than JSON will be supported in later versions. Click Advanced Settings and select the check box next to Table Header: No . If the check box is not selected, the table header is displayed. If the check box is deselected, no table header is displayed.

5.2 Flink Job Management

5.2.1 Flink Job Management

On the Job Management page of Flink jobs, you can submit a Flink job. Currently, the following job types are supported:

- **Flink SQL:** uses SQL statements to define jobs and can be submitted to any queue.
- **Flink Jar:** customizes a JAR package job based on Flink APIs. It can run on dedicated queues.

- **Flink Edge SQL:** analyzes edge device data using SQL and can be deployed on edge nodes.

Flink job management provides the following functions:

- [Managing Flink Job Permissions](#)
- [Creating a Flink SQL Job](#)
- [Creating a User-Defined Flink Job](#)
- [Debugging a Job](#)
- [Editing a job](#)
- [Starting a Job](#)
- [Stopping a Job](#)
- [Deleting a Job](#)
- [Exporting a Job](#)
- [Importing a Job](#)
- [Modifying Name and Description](#)
- [Importing to a Savepoint](#)
- [Triggering a Savepoint](#)
- [Job Details](#)

In addition, you can click **Quick Links** to switch to the details on *User Guide*.

Assigning Agency Permissions

Agencies are required for DLI to execute Flink jobs. You can set the agency when logging in to the management console for the first time or go to **Global Configurations > Service Authorization** to modify the agencies.

The permissions are as follows:

- **Tenant Administrator (global):** **Tenant Administrator** permissions are required to access data from OBS to execute Flink jobs on DLI, for example, obtaining OBS/DWS data sources, log dump (including bucket authorization), checkpointing enabling, and job import and export.

NOTE

Due to cloud service cache differences, operations require about 60 minutes to take effect.

- **DIS Administrator** permissions are required to use DIS data as the data source of DLI Flink jobs.

NOTE

Due to cloud service cache differences, operations require about 30 minutes to take effect.

- To use CloudTable data as the data source of DLI Flink jobs, **CloudTable Administrator** permissions are required.

NOTE

Due to cloud service cache differences, operations require about 3 minutes to take effect.

- To use IEF to execute Flink Edge jobs in DLI, **Tenant Administrator** permissions are required. This permission is also required for running other services that require the **Tenant Administrator** permission.

NOTE

Due to cloud service cache differences, operations require about 3 minutes to take effect.

Job Management page

On the **Overview** page, click **Flink Jobs** to go to the SQL job management page. Alternatively, you can click **Job Management > Flink Jobs**. The **Flink Jobs** page displays all Flink jobs. If there are a large number of jobs, they will be displayed on multiple pages. DLI allows you to view jobs in all statuses.

Table 5-3 Job management parameters

Parameter	Description
ID	ID of a submitted Flink job, which is generated by the system by default.
Name	Name of the submitted Flink job.
Type	Type of the submitted Flink job. Including: <ul style="list-style-type: none">• Flink SQL: Flink SQL job• Flink Jar: Custom Flink job
Status	Job statuses, including: <ul style="list-style-type: none">• Draft• Submitting• Submission failed• Running (The billing starts. After the job is submitted, a normal result is returned.)• Running Exception (The billing stops. If an exception occurs during job running, the job is stopped.)• Stopping• Stopped• Stopping failed• Creating the savepoint• Stopped due to arrears (The billing ends. The job is stopped because the user account is in arrears.)• Restoring (recharged jobs) (The account in arrears is recharged, and the job is being restored).• Completed
Description	Description of the submitted Flink job.
Username	Name of the user who submits a job.

Parameter	Description
Created	Time when a job is created. Jobs can be displayed in ascending or descending order of the job creation time.
Started	Time when a Flink job starts to run.
Duration	Time consumed by job running.
Operation	<ul style="list-style-type: none"> ● Edit: Edit a created job. For details, see Editing a Job. ● Start: Start and run a job. For details, see Starting a Job. ● More <ul style="list-style-type: none"> – FlinkUI: After you click this button, the Flink job execution page is displayed. – Stop: Stop a job in the Submitting or Running status. – Delete: Delete a job. <p>NOTE A deleted job cannot be restored. Therefore, exercise caution when deleting a job.</p> <ul style="list-style-type: none"> – Modify Name and Description: You can modify the name and description of a job. – Import to Save Point: Import the data exported from the original CS job. – Trigger Savepoint: You can click this button for jobs in the Running status to save the job status. – Manage Permissions: You can view the user permissions corresponding to the job and grant permissions to other users.

5.2.2 Managing Flink Job Permissions

Scenarios

- You can isolate Flink jobs allocated to different users by setting permissions to ensure data query performance.
- The administrator and job creator have all permissions, which cannot be set or modified by other users.

Flink Job Permission Operations

1. On the left of the DLI management console, choose **Job Management > Flink Jobs**.
2. Select the job to be configured and choose **More > Manage Permissions** in the **Operation** column. The **User Permission Info** area displays the list of users who have permissions on the job.
You can assign queue permissions to new users, modify permissions for users who have some permissions of a queue, and revoke all permissions of a user on a queue.

- Assign permissions to a new user.
A new user does not have permissions on the job.
 - i. Click **Grant Permission** on the right of **User Permission Info** page. The **Grant Permission** dialog box is displayed.
 - ii. Specify **Username** and select corresponding permissions.
 - iii. Click **OK**.

[Table 5-4](#) describes the related parameters.

Table 5-4 Permission parameters

Parameter	Description
Username	Name of the authorized user. NOTE The username is the name of an existing IAM user.
Select the permissions to be granted to the user	<ul style="list-style-type: none"> ● View Job Details: This permission allows you to view the job details. ● Modify Job: This permission allows you to modify the job. ● Delete Job: This permission allows you to delete the job. ● Start Job: This permission allows you to start the job. ● Stop Job: This permission allows you to stop the job. ● Export Job: This permission allows you to export the job. ● Grant Permission: This permission allows you to grant job permissions to other users. ● Revoke Permission: This permission allows you to revoke the job permissions that other users have but cannot revoke the job creator's permissions. ● View Other User's Permissions: This permission allows you to view the job permissions of other users.

- To assign or revoke permissions of a user who has some permissions on the job, perform the following steps:
 - i. In the list under **User Permission Info** for a job, select the user whose permissions need to be modified and click **Set Permission** in the **Operation** column.
 - ii. In the displayed **Set Permission** dialog box, modify the permissions of the current user. [Table 5-4](#) lists the detailed permission descriptions.
If all options under **Set Permission** are gray, you are not allowed to change permissions on this job. You can apply to the administrator,

job creator, or other authorized users for job permission granting and revoking.

- iii. Click **OK**.
- To revoke all permissions of a user on a job, perform the following steps:
In the list under **User Permission Info** for a job, select the user whose permissions need to be revoked and click **Revoke Permission** in the **Operation** column. After you click **OK** in the **Revoke Permission** dialog box, the user does not have any permission on the job.

Flink Job Permissions

- **View Job Details**
 - Tenants and the admin user can view and operate all jobs.
 - Sub-users and users with the read-only permission can view only their own jobs and the jobs to which the sub-user were granted all permissions.
- **Start Job**
 - To use an exclusive queue, you must have the permission to submit and start jobs.
 - To use a shared queue, you only need to have the permission to start jobs.
- **Stop Job**
 - To use an exclusive queue, you must have the permission to stop jobs and queues.
 - To use a shared queue, you only need to have the permission to stop jobs.
- **Delete Job**
 - If a job can be deleted, you can delete the job if you were granted this permission.
 - If a job cannot be deleted, the system stops the job before you delete it. For details about how to stop a job, see [Stop Job](#). In addition, you must have the permission to delete the job.
- **Create Job**
 - By default, sub-users cannot create jobs.
 - To create a job, you must have this permission. Currently, only the admin user has the permission to create jobs. In addition, the user must have the permission of the related package group or package used by the job.
- **Modify Job**

When modifying a job, you need to have the permission to update the job and the permission to the package group or package used by the job belongs.

5.2.3 Preparing Data

To create a Flink job, you need to enter the data source and data output channel, that is, source and sink. To use another service as the source or sink stream, you need to apply for the service first.

Flink jobs support the following data sources and output channels:

- DIS as the data input and output channel
To use DIS as the data source and output channel, you need to enable DIS first.
For details about how to create a DIS stream, see **Creating a DIS Stream** in the *Data Ingestion Service User Guide*.
After applying for a DIS stream, you can upload local data to DIS to provide data sources for Flink jobs in real time. For details, see **Sending Data to DIS** in the *Data Ingestion Service User Guide*.
An example is provided as follows:

```
1,lilei,bmw320i,28
2,hanmeimei,audia4,27
```
- OBS as the data source
To use OBS as the data source, enable OBS first. For details about how to enable OBS, see **Enabling OBS** in the *Object Storage Service Console Operation Guide*.
After you enable OBS, upload local files to OBS using the Internet. For detailed operations, see **Uploading a File** in the *Object Storage Service Console Operation Guide*.
- RDS as the output channel
To use RDS as the output channel, create an RDS instance. For details, see **Creating a DB Instance** in the *Relational Database Service User Guide*.
- SMN as the output channel
To use SMN as the output channel, create an SMN topic to obtain the URN resource ID and then add topic subscription. For detailed operations, see **Getting Started** in the *Simple Message Notification User Guide*.
- Kafka as the data input and output channel
If Kafka serves as both the source and sink streams, create an enhanced datasource connection between Flink jobs and Kafka. For details, see **Enhanced Datasource Connections**.
If the port of the Kafka server is listened on by the host name, you need to add the mapping between the host name and IP address of the Kafka Broker node to the datasource connection.
- CloudTable as the data input and output channel
To use CloudTable as the data input and output channel, create a cluster in CloudTable and obtain the cluster ID.
- CSS as the output channel
To use CSS as the data output channel, create a cluster in CSS and obtain the cluster's private network address. For details, see **Getting Started** in the *Cloud Search Service User Guide*.
- DCS as the output channel
To use DCS as the output channel, create a Redis cache instance in DCS and obtain the address used for Flink jobs to connect to the Redis instance. For detailed operations, see **Getting Started** in the *Distributed Cache Service User Guide*.

5.2.4 Creating a Flink SQL job

This section describes how to create a Flink streaming SQL job. Flink SQL provides users a method for compiling jobs based on their logic requirements. SQL-based

business logic expression facilitates service implementation. Currently, Flink job supports compiling Flink SQL statements by using the SQL editor and visual editor. This section describes how to use the SQL editor to compile Flink streaming SQL jobs.

Prerequisites

You have prepared the data input and data output channels. For details, see [Preparing Data](#).

Creating a Flink SQL job

Step 1 In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.

Step 2 In the upper right corner of the **Flink Jobs** page, click **Create Job**.

Step 3 Specify job parameters.

Table 5-5 Job configuration information

Parameter	Description
Type	Set Type to Flink SQL . You will need to start jobs by compiling SQL statements.
Name	Name of a job. Enter 1 to 57 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. NOTE The job name must be globally unique.
Description	Description of a job. It can be up to 512 characters long.
Editor	SQL Editor and Visual Editor are both available. SQL Editor is used by default.
Template Name	This parameter is only used when Editor is set to SQL Editor . You can select a sample template or a custom job template. For details about templates, see Flink Template Management .

Step 4 Click **OK** to enter the **Edit** page.

Step 5 Edit a Flink SQL job

Enter details SQL statements in the SQL statement edit area. For details about SQL statements, see the *Data Lake Insight SQL Syntax Reference*.

Step 6 Click **Check Semantics**.

- You can **Debug** or **Start** a job only after the semantic verification is successful.
- If verification is successful, the message "The SQL semantic verification is complete. No error." will be displayed.

- If verification fails, a red "X" mark will be displayed in front of each SQL statement that produced an error. You can move the cursor to the "X" mark to view error details and change the SQL statement as prompted.

Step 7 Set job running parameters.**Table 5-6** Running parameters

Parameter	Description
CUs	Sum of the number of compute units and job manager CUs of DLI. One CU equals one vCPU and 4 GB.
Job Manager CUs	Number of CUs of the management unit.
Max Concurrent Jobs	Maximum number of Flink SQL jobs that run at the same time. NOTE This value cannot be greater than four times the compute units (number of CUs minus the number of job manager CUs).
Task Manager Configuration	Whether to set Task Manager resource parameters If this option is selected, you need to set the following parameters: <ul style="list-style-type: none">• CU(s) per TM: Number of resources occupied by each Task Manager.• Slot(s) per TM: Number of slots contained in each Task Manager.
Save Job Log	Whether to save the job running logs to OBS If this option is selected, you need to set the following parameters: OBS Bucket: Select an OBS bucket to store user job logs. If the selected OBS bucket is not authorized, click Authorize . NOTE If both Enable Checkpointing and Save Job Log are selected, you only need to authorize OBS once.
Alarm Generation upon Job Exception	Whether to report job exceptions, for example, abnormal job running or exceptions due to an insufficient balance, to users via SMS or email If this option is selected, you need to set the following parameters: SMN topic Select a user-defined SMN topic. For details about how to customize SMN topics, see Creating a Topic in the <i>Simple Message Notification User Guide</i> .

Parameter	Description
Enable Checkpointing	<p>Whether to enable job snapshots. If this function is enabled, jobs can be restored based on the checkpoints.</p> <p>If this option is selected, you need to set the following parameters:</p> <ul style="list-style-type: none">• Checkpoint Interval is the number of seconds between when new checkpoints are created. The parameter value ranges from 1 to 999999, and the default value is 10.• Checkpoint Mode can be set to either of the following values:<ul style="list-style-type: none">– At least once: Events are processed at least once.– Exactly once: Events are processed only once.• OBS Bucket: Select an OBS bucket to store your checkpoints. If the selected OBS bucket is not authorized, click Authorize. <p>NOTE If both Enable Checkpointing and Save Job Log are selected, you only need to authorize OBS once.</p>
Auto Restart upon Exception	<p>Whether to enable automatic restart. If this function is enabled, any job that has become abnormal will be automatically restarted.</p> <p>If this option is selected, you need to set the following parameters:</p> <p>Restore Job from Checkpoint: This parameter is available only when Enable Checkpointing is selected.</p>
Idle State Retention Time	<p>Defines for how long the state of a key is retained without being updated before it is removed in GroupBy or Window. The default value is 1 hour.</p>
Dirty Data Policy	<p>Select a policy for processing dirty data. The following policies are supported: Ignore, Trigger a job exception, and Save.</p> <p>NOTE Save indicates that the dirty data is stored to the OBS bucket selected above.</p>
Dirty Data Dump Address	<p>Set this parameter when Dirty data policy is set to Save. Click the address box to select the OBS path for storing dirty data.</p>


Parameter	Description
Queue	<p>Shared queue is selected by default. You can also select a dedicated custom queue.</p> <p>If you selected custom queues, you need to configure the following parameters:</p> <p>UDF Jar: Before selecting a JAR file to be inserted, upload the corresponding JAR package to the OBS bucket and choose Data Management > Package Management to create a package. For details, see Creating a Package.</p> <p>In SQL, you can call a user-defined function that is inserted into a JAR file.</p> <p>NOTE</p> <p>During job creation, a sub-user can only select a queue that has been allocated to the user.</p> <p>If the remaining capacity of the selected queue cannot meet the job requirements, the system automatically scales up the capacity and you will be billed based on the increased capacity. When a queue is idle, the system automatically scales in the queue.</p>
UDF Jar	User-defined UDF files can be uploaded to DLI using the Package Management function.

Step 8 Click **Save**.

Step 9 Click **Start**. On the displayed **Start Flink Job** page, confirm the job specifications, and click **Start Now** to start the job.

After the job is started, the system automatically switches to the **Flink Jobs** page, and the created job is displayed in the job list. You can view the job status in the **Status** column. After a job is successfully submitted, the job status will change from **Submitting** to **Running**. After the execution is complete, the message **Completed** is displayed.

If the job status is **Submission failed** or **Running exception**, the job submission failed or the job did not execute successfully. In this case, you can move the cursor over the status icon in the **Status** column of the job list to view the error details.

You can click  to copy these details. After handling the fault based on the provided information, resubmit the job.

 **NOTE**

Other buttons are as follows:

- **Save As:** Save the created job as a new job.
- **Debug:** Perform job debugging. For details, see [Debugging a Job](#).
- **Format:** Format the SQL statements in the editing box.
- **Set as Template:** Set the created SQL statements as a job template.
- **Theme Settings:** Set the theme related parameters, including **Font Size**, **Wrap**, and **Page Style**.

----End

5.2.5 Creating a Custom Flink Job

This section describes how to create a custom Flink job. You can perform secondary development based on Flink APIs, build your own JAR file, and submit the JAR file to DLI queues. DLI is fully compatible with open-source community APIs. To create a custom Flink job, you need to compile and build application JAR files. You must have a certain understanding of Flink secondary development and have high requirements related to stream computing complexity.

Prerequisites

- Ensure that a dedicated queue has been created.
- When running a custom Flink job, you need to build the secondary development application code into a JAR file and upload the JAR file to the created OBS bucket. Choose **Data Management > Package Management** to create a package. For details, see [Creating a Package](#).

NOTE

DLI does not support the download function. If you need to modify the uploaded data file, please edit the local file and upload it again.

- Flink dependencies have been built in the DLI server and security hardening has been performed based on the open-source community version. To avoid dependency package compatibility issues or log output and dump issues, be careful to exclude the following files when packaging:
 - Built-in dependencies (or set the package dependency scope to "provided")
 - Log configuration files (example, log4j.properties/logback.xml)
 - JAR package for log output implementation (example, log4j).

Creating a Custom Flink Job

Step 1 In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.

Step 2 In the upper right corner of the **Flink Jobs** page, click **Create Job**.

Step 3 Configure job information.

Table 5-7 Job configuration information

Parameter	Description
Type	Select Flink Jar .
Name	Name of a job. Enter 1 to 57 characters. Only letters, digits, hyphens (-), and underscores (_) are allowed. NOTE The job name must be globally unique.
Description	Description of a job. It can be up to 512 characters long.

Step 4 Click **OK** to enter the **Edit** page.

Step 5 Upload the JAR file.

Table 5-8 Description

Name	Description
Application	User-defined package. Before selecting a JAR file to be inserted, upload the corresponding JAR file to the OBS bucket and choose Data Management > Package Management to create a package. For details, see Creating a Package .
Main Class	The name of the JAR package to be loaded, for example, KafkaMessageStreaming . <ul style="list-style-type: none">• Default: Specified based on the Manifest file in the JAR package.• Manually assign: Enter the class name and confirm the class arguments (separate arguments with spaces). NOTE When a class belongs to a package, the package path must be carried, for example, packageName.KafkaMessageStreaming .
Class Arguments	List of arguments of a specified class. The arguments are separated by spaces.
JAR Package Dependencies	Select a user-defined package dependency. Before selecting a JAR file to be inserted, upload the corresponding JAR file to the OBS bucket and choose Data Management > Package Management to create a package. Select JAR as the package type. For details, see Creating a Package .
Other Dependencies	User-defined dependency files. Before selecting a dependency file, upload the file to the OBS bucket and choose Data Management > Package Management to create a package. The package type is not limited. For details, see Creating a Package . You can add the following content to the application to access the corresponding dependency file: In the command, fileName indicates the name of the file to be accessed, and ClassName indicates the name of the class that needs to access the file. <code>ClassName.class.getClassLoader().getResource("userData/fileName")</code>
Image	Image set on the Software Repository for Container (SWR) console.

Step 6 Configure job parameters.

Table 5-9 Description

Name	Description
CUs	One CU has one vCPU and 4-GB memory. The number of CUs ranges from 2 to 400.
Job Manager CUs	Set the number of CUs on a management unit. The value ranges from 1 to 4. The default value is 1.
Max Concurrent Jobs	Maximum number of parallel operators in a job. NOTE <ul style="list-style-type: none">The value must be less than or equal to four times the number of compute units (CUs minus the number of job manager CUs).You are advised to set this parameter to a value greater than that configured in the code. Otherwise, job submission may fail.
Task Manager Configuration	Whether to set Task Manager resource parameters If this option is selected, you need to set the following parameters: <ul style="list-style-type: none">CU(s) per TM: Number of resources occupied by each Task Manager.Slot(s) per TM: Number of slots contained in each Task Manager.
Save Job Log	Whether to save the job running logs to OBS If this option is selected, you need to set the following parameters: OBS Bucket: Select an OBS bucket to store user job logs. If the selected OBS bucket is not authorized, click Authorize .
Alarm Generation upon Job Exception	Whether to report job exceptions, for example, abnormal job running or exceptions due to an insufficient balance, to users via SMS or email If this option is selected, you need to set the following parameters: SMN Topic Select a user-defined SMN topic. For details about how to customize SMN topics, see Creating a Topic in the <i>Simple Message Notification User Guide</i> .
Auto Restart upon Exception	Whether to enable automatic restart. If this function is enabled, any job that has become abnormal will be automatically restarted.

Step 7 Select a queue to which the job belongs.

 **NOTE**


- A custom Flink job can run only on a created dedicated queue.
- If no dedicated queue is available in the **Queue** drop-down list, create a dedicated queue and bind it to the current user.

Step 8 Click **Save** on the upper right of the page.

Step 9 Click **Start** on the upper right side of the page. On the displayed **Start Flink Job** page, confirm the job specifications, and click **Start Now** to start the job.

After the job is started, the system automatically switches to the **Flink Jobs** page, and the created job is displayed in the job list. You can view the job status in the **Status** column. After a job is successfully submitted, the job status will change from **Submitting** to **Running**. After the execution is complete, the message **Completed** is displayed.

If the job status is **Submission failed** or **Running exception**, the job submission failed or the job did not execute successfully. In this case, you can move the cursor over the status icon in the **Status** column of the job list to view the error details.

You can click  to copy these details. After handling the fault based on the provided information, resubmit the job.

NOTE

Other buttons are as follows:

Save As: Save the created job as a new job.

----End

5.2.6 Debugging a Job

The job debugging function helps you check the logic correctness of your compiled SQL statements before running a job.

NOTE

- Currently, only Flink SQL jobs support this function.
- The job debugging function is used only to verify the SQL logic and does not involve data write operations.

Procedures

Step 1 In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.

Step 2 In the **Operation** column of the created Flink SQL job, click **Edit**. The page for editing the Flink SQL job is displayed.

For a job that is being created, you can debug the job on the **Edit** page.

Step 3 Click **Debug** above the SQL editing box of the **Edit** page to parse the edited SQL statement. The **Debugging Parameters** page is displayed in the right pane of the **Edit** page.

- **Dump Bucket:** Select an OBS bucket to save debugging logs. If you select an unauthorized OBS bucket, click **Authorize**.
- **Data Input Mode:** You can select CSV data stored in the OBS bucket or manually enter the data.
 - **OBS (CSV)**

If you select this value, prepare OBS data first before using DLI. For details, see [Preparing Data](#). OBS data is stored in CSV format, where

multiple records are separated by line breaks and different fields in a single record are separated by commas (,). In addition, you need to select a specific object in OBS as the input source data.

– **Manual typing**

If you select this value, compile SQL statements as data sources. In this mode, you need to enter the value of each field in a single record.

Step 4 Click **Start Debugging**. Once debugging is complete, the **Debugging Result** page appears.

- If the debugging result meets the expectation, the job is running properly.
- If the debugging result does not meet the expectation, business logic errors may have occurred. In this case, modify SQL statements and conduct debugging again.

----End

5.2.7 Performing Operations on a Job

After a job is created, you can perform operations on the job as required.

- [Editing a Job](#)
- [Starting a Job](#)
- [Stopping a job](#)
- [Deleting a Job](#)
- [Exporting a Job](#)
- [Importing a Job](#)
- [Modifying Name and Description](#)
- [Importing to a Savepoint](#)
- [Triggering a Savepoint](#)

Editing a Job

You can edit a created job, for example, by modifying the SQL statement, job name, job description, or job configurations.

Step 1 In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.

Step 2 In the row where the job you want to edit is located, click **Edit** in the **Operation** column to switch to the **Edit** page.

Step 3 Edit the job as required.

For details about how to edit a Flink SQL job, see Step 5 to Step 7 in [Creating a Flink SQL Job](#).

For details about how to edit a user-defined Flink job, see Step 5 to Step 7 in [Creating a User-Defined Flink Job](#).

----End

Starting a Job

You can start a saved or stopped job.

Step 1 In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.

Step 2 Use either of the following methods to start jobs:

- **Starting a single job**
Select a job and click **Start** in the **Operation** column.
Alternatively, you can select the row where the job you want to start resides and click **Start** in the upper left of the job list.
- **Batch starting jobs**
Select the rows where the jobs you want to start reside and click **Start** in the upper left of the job list.

After you click **Start**, the **Start Flink Job** page is displayed.

Step 3 On the **Start Flink Job** page, confirm the job information. If they are correct, click **Start Now**.

After a job is started, you can view the job execution result in the **Status** column.

----End

Stopping a job

You can stop a job in the **Running** or **Submitting** state.

Step 1 In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.

Step 2 Stop a job using either of the following methods:

- **Stopping a job**
Locate the row that contains the job to be stopped, click **More** in the **Operation** column, and select **Stop**.
Alternatively, you can select the row where the job you want to stop resides and click **Stop** in the upper left of the job list.
- **Batch stopping jobs**
Locate the rows where the jobs you want to stop reside and click **Stop** in the upper left of the job list.

Step 3 In the displayed **Stop Job** dialog box, click **OK** to stop the job.

NOTE

- Before stopping a job, you can trigger a savepoint to save the job status information. When you start the job again, you can choose whether to restore the job from the savepoint.
- If you select **Trigger savepoint**, a savepoint is created. If **Trigger savepoint** is not selected, no savepoint is created. By default, the savepoint function is disabled.
- The lifecycle of a savepoint starts when the savepoint is triggered and stops the job, and ends when the job is restarted. The savepoint is automatically deleted after the job is restarted.

When a job is being stopped, the job status is displayed in the **Status** column of the job list. The details are as follows:

- **Stopping**: indicates that the job is being stopped.
- **Stopped**: indicates that the job is stopped successfully.
- **Stop failed**: indicates that the job failed to be stopped.

----End

Deleting a Job

If you do not need to use a job, perform the following operations to delete it. A deleted job cannot be restored. Therefore, exercise caution when deleting a job.

Step 1 In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.

Step 2 Perform either of the following methods to delete jobs:

- Deleting a single job

Locate the row where the job you want to delete resides and click **More > Delete** in the **Operation** column.

Alternatively, you can select the row where the job you want to delete resides and click **Delete** in the upper left of the job list.

- Deleting jobs in batches

Select the rows where the jobs you want to delete reside and click **Delete** in the upper left of the job list.

Step 3 Click **OK**.

----End

Exporting a Job

You can export the created Flink jobs to an OBS bucket.

This mode is applicable to the scenario where a large number of jobs need to be created when you switch to another region, project, or user. In this case, you do not need to create a job. You only need to export the original job, log in to the system in a new region or project, or use a new user to import the job.

NOTE

When switching to another project or user, you need to grant permissions to the new project or user. For details, see [Managing Flink Job Permissions](#).

Step 1 In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.

Step 2 Click **Export Job** in the upper right corner. The **Export Job** dialog box is displayed.

Step 3 Select the OBS bucket where the job is stored. Click **Next**.

Step 4 Select Job

By default, configurations of all jobs are exported. You can enable the **Custom Export** function to export configurations of the desired jobs.

Step 5 Click **Confirm** to export the job.

----End

Importing a Job

You can import the Flink job configuration file stored in the OBS bucket to the **Flink Jobs** page of DLI.

This mode is applicable to the scenario where a large number of jobs need to be created when you switch to another region, project, or user. In this case, you do not need to create a job. You only need to export the original job, log in to the system in a new region or project, or use a new user to import the job.

If you need to import a self-created job, you are advised to use the job creation function. For details, see [Creating a Flink SQL job](#) and [Creating a Custom Flink Job](#).

NOTE

- When switching to another project or user, you need to grant permissions to the new project or user. For details, see [Managing Flink Job Permissions](#).
- Only jobs whose data format is the same as that of Flink jobs exported from DLI can be imported.

Step 1 In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.

Step 2 Click **Import Job** in the upper right corner. The **Import Job** dialog box is displayed.

Step 3 Select the OBS bucket where the job configuration file to be imported is located. Click **Next**.

Step 4 Configure Same-Name Job Policy Click **Next**.

- Select **Overwrite job of the same name**. If the name of the job to be imported already exists, the existing job configuration will be overwritten and the job status switches to **Draft**.
- If **Overwrite job of the same name** is not selected and the name of the job to be imported already exists, the job will not be imported.

Step 5 Ensure that **Select Configuration File** and **Configure Same-Name Job Policy** are correctly configured. Click **Confirm** to export the job.

----End

Modifying Name and Description

You can change the job name and description as required.

Step 1 In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.

Step 2 In the **Operation** column of the job whose name and description need to be modified, choose **More > Modify Name and Description**. The **Modify Name and Description** dialog box is displayed. Change the name or modify the description of a job.

Step 3 Click **OK**.

----End

Importing to a Savepoint

When migrating a job from CS to DLI, you can import the job data by importing to a savepoint.

Triggering a Savepoint

You can click **Trigger Savepoint** for jobs in the **Running** status to save the job status.

5.2.8 Job Details

After creating a job, you can view the job details to learn about the following information:

- [Viewing Job Details](#)
- [Checking the Job Monitoring Information](#)
- [Viewing Job Running Logs](#)

Viewing Job Details

This section describes how to view job details. After you create and run a job, you can view job details, including SQL statements and parameter settings. For a user-defined job, you can only view its parameter settings.

Step 1 In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.

Step 2 Click the name of the job to be viewed. The **Job Details** page is displayed.

On the **Job Details** page, you can view SQL statements, configured parameters.

The following uses a Flink SQL job as an example.

Table 5-10 Description

Parameter	Description
Job Status	The current status of the job is displayed on the right of the job name.
ID	Job ID.
Job Type	Job type, for example, Flink SQL.
Queue	If the queue to which the job belongs is a shared queue, the shared queue is displayed. If the queue to which the job belongs is a user-defined dedicated queue, the queue name is displayed.

Parameter	Description
Running Mode	If you create a job in a shared queue, this parameter is Shared . If you create a job in a user-defined queue, this parameter is Exclusive .
CUs	Number of CUs configured for a job.
Job Manager CUs	Number of job manager CUs configured for a job.
Max Concurrent Jobs	Maximum number of jobs that can be concurrently executed by a Flink job.
Save Job Log	Enabled or disabled.
Auto Restart upon Exception	Enabled or disabled.
Restore Job from Checkpoint	Enabled or disabled.
CU(s) per TM	Number of CUs occupied by each Task Manager configured for a job.
Slot(s) per TM	Number of Task Manager slots configured for a job.
Enable Checkpointing	Enabled or disabled.
Checkpoint Interval	Interval between storing intermediate job running results to OBS.
Checkpoint Mode	Checkpoint mode. Values include: <ul style="list-style-type: none">• At least once: Events are processed at least once.• Exactly once: Events are processed only once.
Idle State Retention Time	Defines for how long the state of a key is retained without being updated before it is removed in GroupBy or Window.

----End

Checking the Job Monitoring Information

You can use Cloud Eye to view details about job data input and output.

Step 1 In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.

Step 2 Click the name of the job to be viewed. The **Job Details** page is displayed.

Click **Job Monitoring** in the upper right corner of the page to switch to the Cloud Eye console.

The following table describes monitoring metrics related to Flink jobs.

Table 5-11 Monitoring metrics related to Flink jobs

Name	Description
Flink job data read rate	Displays the data input rate of a Flink job for monitoring and debugging. Unit: record/s.
Flink job data write rate	Displays the data output rate of a Flink job for monitoring and debugging. Unit: record/s.
Flink job total data read	Displays the total number of data inputs of a Flink job for monitoring and debugging. Unit: records
Flink job total data write	Displays the total number of output data records of a Flink job for monitoring and debugging. Unit: records
Flink job byte read rate	Displays the number of input bytes per second of a Flink job. Unit: byte/s
Flink job byte write rate	Displays the number of output bytes per second of a Flink job. Unit: byte/s
Flink job total read byte	Displays the total number of input bytes of a Flink job. Unit: byte
Flink job total write byte	Displays the total number of output bytes of a Flink job. Unit: byte
Flink job CPU usage	Displays the CPU usage of Flink jobs. Unit: %
Flink job memory usage	Displays the memory usage of Flink jobs. Unit: %

----End

Viewing Job Running Logs

You can view the run logs to locate the faults occurring during job running.

- Step 1** In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.
- Step 2** Click the name of the job to be viewed. The **Job Details** page is displayed.
- Step 3** On the **Run Log** tab page, you can view the **Job Manager** and **Task Manager** information of the running job.

Information about Job Manager and Task Manager is updated every minute. By default, only the run logs generated within the last 1 minute are displayed. You can click **Log history** to view more logs.

If you select an OBS bucket for saving job logs during the job configuration, you can switch to the OBS bucket and download log files to view more historical logs.

If the job is not running, information on the **Task Manager** page cannot be viewed.

----End

5.3 Spark Job Management

5.3.1 Spark Job Management

Spark job management provides the following functions:

- [Creating a Spark Job](#)
- [Re-executing a Job](#)
- [Searching for a Job](#)
- [Terminating a Job](#)

In addition, you can click **Quick Links** to switch to the details on *User Guide*.

Job Management Page

On the **Overview** page, click **Spark Jobs** to go to the SQL job management page. Alternatively, you can click **Job Management > Spark Jobs**. The **Spark Jobs** page displays all Spark jobs. If there are a large number of jobs, they will be displayed on multiple pages. DLI allows you to view jobs in all statuses.

 **NOTE**

After the job is executed successfully, the job record is saved for only 6 hours.

Table 5-12 Job management parameters

Parameter	Description
Job ID	ID of a submitted Spark job, which is generated by the system by default.
Job Name	Name of a submitted Spark job.
Executed By	Name of the user who executed the Spark job.
Job Status	Job status. The following values are available: <ul style="list-style-type: none">• Starting: The job is being started.• Running: The job is being executed.• Failed: The session has exited.• Finished: The session is successfully executed.• Restoring: The job is being restored.
Queue	Queue where the submitted Spark job is located.
Created	Time when a job is created. Jobs can be displayed in ascending or descending order of the job creation time.
Last Modified	Time when a job is completed.

Parameter	Description
Operation	<ul style="list-style-type: none">● Edit: You can modify the current job configuration and re-execute the job.● SparkUI: After you click this button, the Spark job execution page is displayed. NOTE The SparkUI page cannot be viewed for jobs in the Starting state.● Terminate Job: Cancel the current job.● Archive Log: Save job logs to the temporary bucket created by DLI.● Export Log: Export logs to the created OBS bucket. NOTE<ul style="list-style-type: none">● You have the permission to create OBS buckets.● If the job is in the Running state, logs cannot be exported.● Commit Log: View the logs of submitted jobs.● Driver Log: View the logs of running jobs.

Re-executing a Job

On the **Spark Jobs** page, click **Edit** in the **Operation** column of the job. On the Spark job creation page that is displayed, modify parameters as required and execute the job.

Searching for a Job

On the **Spark Jobs** page, select **Job Status** or **Queue**. The system displays the jobs that meet the filter condition in the job list.

Terminating a Job

On the **Spark Jobs** page, choose **More > Terminate Job** in the **Operation** column of the job to stop the job.

Exporting Logs

On the **Spark Jobs** page, choose **More > Export Log** in the **Operation** column of the corresponding job. In the dialog box that is displayed, enter the path of the created OBS bucket and click **OK**.

5.3.2 Creating a Spark Job

DLI provides fully-managed Spark computing services by allowing you to execute Spark jobs.

On the **Overview** page, click **Create Job** in the upper right corner of the **Spark Jobs** tab or click **Create Job** in the upper right corner of the **Spark Jobs** page. The Spark job editing page is displayed.

Enter the Spark job editing page. A message is displayed, indicating that a temporary DLI data bucket will be created. The created bucket is used to store

temporary data generated by DLI, such as job logs and job results. You cannot view job logs if you choose not to create it. You can periodically delete objects in a bucket or transit objects between different storage classes. The bucket name is set by default.

If you do not need to create a DLI temporary data bucket and do not want to receive this message, select **Do not show again** and click **Cancel**.






Prerequisites

Upload the dependencies to the corresponding OBS bucket on the **Data Management > Package Management** page. For details, see [Creating a Package](#).

GUI Description

- Navigation bar on the left
On the Spark job creation page, the navigation tree on the left contains the **Queues** and **Packages** tab pages.

Table 5-13 Description of buttons in the left navigation pane

No.	Tab/Button	Tab/Button Name	Description
1		Queues	Displays existing queues.
2		Packages	Displays existing packages.
3		Create	Create a queue or a package.
4		Refresh	Refreshes the lists of existing queues and packages.
5		Search	On the Packages tab page, enter a package name for search.

- Job editing window
In the job editing window, you can set parameters in **Fill Form** mode or **Write API** mode.
The following uses the **Fill Form** as an example. In **Write API** mode, refer to the *Data Lake Insight API Reference* for parameter settings.
 - **Select a Queue:** For details about the parameters, see [Table 5-14](#).

Table 5-14 Queue parameters

Parameter	Description
Queue	Select the target queue from the drop-down list box.

- **Job Configurations:** Refer to [Table 5-15](#) for details.

Table 5-15 Job configuration parameters

Parameter	Description
Job Name	Set a job name.
Application	Select the package to be executed. The value can be .jar or .py .
Main Class	Enter the name of the main class. When the application type is .jar , the main class name cannot be empty.
Application Parameters	User-defined parameter. Separate multiple parameters by Enter .
Spark Arguments	Enter a parameter in the format of key=value . Press Enter to separate multiple key-value pairs.
JAR Package Dependencies	JAR file on which the Spark job depends
Python File Dependencies	py-files on which the Spark job depends
Other Dependencies	Other files on which the Spark job depends
Group Name	If you select a group when creating a package, you can select all the packages and files in the group. For details about how to create a package, see Creating a Package .
Retry	Indicates whether to retry a failed job. If you select Yes , you need to set the following parameters: Maximum Retries: Maximum number of retry times. The maximum value is 100 .

Parameter	Description
Advanced Settings	<ul style="list-style-type: none"> • Skip • Configure <ul style="list-style-type: none"> - Select Dependency Resources: For details about the parameters, see Table 5-16. - Configure Resources: For details about the parameters, see Table 5-17.

Table 5-16 Parameters for selecting dependency resources

Parameter	Description
Module Name	<p>Dependency modules provided by DLI for executing datasource connection jobs. To access different services, you need to select different modules.</p> <ul style="list-style-type: none"> • CloudTable/MRS HBase: sys.datasource.hbase • CloudTable/MRS OpenTSDB: sys.datasource.opentsdb • RDS MySQL: sys.datasource.rds • RDS PostGre: sys.datasource.rds • DWS: sys.datasource.dws • CSS: sys.datasource.css
Resource Package	JAR package on which the Spark job depends.

Table 5-17 Resource specification parameters

Parameter	Description
Resource Specifications	<p>Select a resource specification from the drop-down list box. The system provides three resource specifications for you to select. The following configuration items in the resource specifications can be modified:</p> <ul style="list-style-type: none"> • Executor Memory • Executor Cores • Executors • Driver Cores • Driver Memory <p>If modified, your modified settings of the items are used.</p>

Parameter	Description
Executor Memory	Customize the configuration item based on the selected resource specifications.
Executor Cores	Customize the configuration item based on the selected resource specifications.
Executors	Customize the configuration item based on the selected resource specifications.
Driver Cores	Customize the configuration item based on the selected resource specifications.
Driver Memory	Customize the configuration item based on the selected resource specifications.

 NOTE

Spark job parameter calculation:

- Number of CUs = Driver Cores + Executors
- Memory = Driver Memory + (Executors x Executor Memory)

Creating a Spark Job

1. In the Spark job editing window, set related parameters. For details, see the description of the Spark job editing window.
2. Click **Execute** in the upper right corner of the Spark job editing window to submit the job. The message "The batch job is submitted." is displayed.
3. (Optional) Go to the **Spark Jobs** page to view the status and logs of the submitted Spark job.

 NOTE

After the job is executed successfully, the job record is saved for only 6 hours.

6 Queue Management

6.1 Overview

Queue

Queues in DLI are computing resources, which are the basis for using DLI. All executed jobs require computing resources.

Currently, DLI provides two types of queues, **For SQL** and **For general use**. SQL queues are used to run SQL jobs. General-use queues are compatible with Spark queues of earlier versions and are used to run Spark and Flink jobs.

 **NOTE**

The SQL queue automatically restarts every 12 hours.

Difference Between Computing and Storage Resources

Table 6-1 Difference between computing and storage resources

Resource	How to Obtain	Function
Compute resource	Create queue on the DLI management console.	Used for executing queries.
Storage resource	DLI has a 5 GB quota.	Used for storing data in the database and DLI table.

 NOTE

- Storage resources are internal storage resources of DLI and represent the amount of data stored in DLI.
- By default, DLI provides a 5 GB quota for storage resources.
- A queue named **default** is preset in DLI. If you are uncertain about the required queue capacity or have no available queue capacity to run queries, you can execute jobs using this queue.
- The **default** queue is used only for user experience. It may be occupied by multiple users at a time. Therefore, it is possible that you fail to obtain the resource for related operations. You are advised to use a self-built queue to execute jobs.

Dedicated Queue

Resources of a dedicated queue are not released when the queue is idle. That is, resources are reserved regardless of whether the queue is used. Dedicated queues ensure that resources exist when jobs are submitted.

Cross-AZ Queues

An AZ contains one or more physical data centers. Each AZ has independent cooling, fire extinguishing, moisture-proofing, and electricity facilities. Within an AZ, computing, network, storage, and other resources are logically divided into multiple clusters. AZs within a region are interconnected using high-speed optical fibers to allow you to build cross-AZ high-availability systems.

DLI dual-AZ queues improve data availability by creating a duplicate queue in the second AZ. You can continuously use DLI when one AZ is unavailable. The dual-AZ mode suits those who require high availability.

A DLI dual-AZ queue is created with the same compute resources in two AZs. For example, if you require 1,400 CUs, you can select 1,400 CUs and select the dual-AZ option when creating a queue. Then, DLI creates 1,400 CUs dedicated compute resources in two AZs. When one AZ is unavailable, the other AZ can properly process your compute tasks.

 NOTE

- Currently, only SQL queues are supported.
- The default queue and are not supported.
- If you select **Dual-AZ** when creating a queue, the billing is twice as that in single-AZ mode.

AI-enhanced queue

AI-enhanced queues support SQL jobs specific to AI analysis. DLI uses AI images to integrate a series of common AI algorithm packages based on the original Spark images. Spark and AI capabilities can be combined, helping you run machine learning jobs. For details about the installation packages on which the AI image depends, see [Dependency Packages for AI Image Installation](#). When creating a queue, select **AI-enhanced** for **Queue Type**.

Modifying Queue Specifications (Scaling)

After purchasing a queue, queue resources may be insufficient or wasted as the service scenario changes.

DLI allows you to scale in or out queue specifications. There are two adjustment modes: manual and scheduled adjustment.

- **Manual scale-out/scale-in:** You can manually change the queue specifications based on service requirements. For details, see [Modifying Queue Specifications \(Manual Scale-out/Scale-in\)](#).
- **Scheduled scale-out/scale-in:** You can set the time for automatic queue scaling based on service requirements. The system periodically triggers queue scaling. For details, see [Scheduling CU Changes \(Periodic Scale-out/Scale-in\)](#).

In addition, DLI can automatically trigger scale-in and scale-out for Flink jobs based on the job size.

NOTE

Scaling can be performed for a newly created queue only when there are jobs running on this queue.

Queue Management Page

Queue Management provides the following functions:

- [Managing Permissions](#)
- [Creating a Queue](#)
- [Deleting a Queue](#)
- [Modifying CIDR Block](#)
- [Modifying Queue Specifications \(Manual Scale-out/Scale-in\)](#)
- [Modifying Queue Specifications \(Scheduled Scale-out/Scale-in\)](#)
- [Creating a Topic for Key Event Notifications](#)

NOTE

To receive notifications when a DLI job fails, **SMN Administrator** permissions are required.

The queue list displays all queues created by you and the **default** queue. Queues are listed in chronological order by default in the queue list, with the most recently created queues displayed at the top.

Table 6-2 Parameter description

Parameter	Description
Name	Name of a queue.
Type	Indicates the queue type. <ul style="list-style-type: none">• For SQL• For general purpose• Spark queue (compatible with earlier versions)

Parameter	Description
Queue Specifications	Queue size. Unit: CU CU is the pricing unit of queues. A CU consists of 1 vCPU and 4-GB memory. The computing capabilities of queues vary with queue specifications. The higher the specifications, the stronger the computing capability.
Owner	Queue owner.
Created	Time when the queue is created.
Description	Description of a queue specified during queue creation. If no description is provided, -- is displayed.
Operation	<ul style="list-style-type: none">● Details: Allow you to view the permission information of the queue and assign queue permissions to users.● Restart: Forcibly restart a queue. NOTE Only the SQL queue has the Restart operation.● Delete: Allow you to delete the selected queue. You cannot delete a queue where there are running jobs or jobs are being submitted.● Modifying CIDR Block: When DLI enhanced datasource connection is used, the CIDR block of the DLI queue cannot overlap with that of the data source. You can modify the CIDR block as required.● Modify Queue Specifications: You can select Scale-out or Scale-in as required. The number of CUs after modification must be an integer multiple of 16.● Create Periodic Task: You can set different queue sizes at different time or in different periods based on the service period or usage. The system automatically performs scale-out or scale-in as scheduled. The target value must be an integral multiple of 16 CUs.● Manage Permissions: You can view the user permissions corresponding to the queue and grant permissions to other users.● Test Address Connectivity: Test whether the queue is reachable to the specified address. The domain name and IP address are supported. The port can be specified.

6.2 Queue Permission Management

Scenarios

- You can isolate queues allocated to different users by setting permissions to ensure data query performance.

- The administrator and queue owner have all permissions, which cannot be set or modified by other users.

Operations

Step 1 On the top menu bar of the DLI management console, click **Queue Management**.

Step 2 Select the queue to be configured and choose **More > Manage Permissions** in the **Operation** column. The **User Permission Info** area displays the list of users who have permissions on the queue.

You can assign queue permissions to new users, modify permissions for users who have some permissions of a queue, and revoke all permissions of a user on a queue.

- Assign permissions to a new user.
A new user does not have permissions on the queue.
 - a. Click **Grant Permission** on the right of **User Permission Info** page. The **Grant Permission** dialog box is displayed.
 - b. Specify **Username** and select corresponding permissions.
 - c. Click **OK**.

Table 6-3 describes the related parameters.

Table 6-3 Parameter description

Parameter	Description
Username	Name of the authorized user. NOTE The username is an existing IAM user name and has logged in to the DLI management console.

Parameter	Description
Permission Settings	<ul style="list-style-type: none">▪ Delete Queues: This permission allows you to delete the queue.▪ Submit Jobs: This permission allows you to submit jobs using this queue.▪ Terminate Jobs: This permission allows you to terminate jobs submitted using this queue.▪ Grant Permission: This permission allows you to grant queue permissions to other users.▪ Revoke Permission: This permission allows you to revoke the queue permissions that other users have but cannot revoke the queue owner's permissions.▪ View Other User's Permissions: This permission allows you to view the queue permissions of other users.▪ Restart Queues: This permission allows you to restart queues.▪ Modify Queue Specifications: This permission allows you to modify queue specifications.

- To assign or revoke permissions of a user who has some permissions on the queue, perform the following steps:
 - a. In the list under **User Permission Info** for a queue, select the user whose permissions need to be modified and click **Set Permission** in the **Operation** column.
 - b. In the displayed **Set Permission** dialog box, modify the permissions of the current user. [Table 6-3](#) lists the detailed permission descriptions.

If all options under **Set Permission** are gray, you are not allowed to change permissions on this queue. You can apply to the administrator, queue owner, or other authorized users for queue permission granting and revoking.
 - c. Click **OK**.
- To revoke all permission of a user on the queue, perform the following steps:

In the user list under **Permission Info**, select the user whose permission needs to be revoked and click **Revoke Permission** under **Operation**. In the **Revoke Permission** dialog box, click **OK**. All permissions on this queue are revoked.

----End

6.3 Creating a Queue

Before executing a job, you need to create a queue.

 NOTE

- If you use a sub-account to create a queue for the first time, log in to the DLI management console using the main account and keep records in the DLI database before creating a queue.
- It takes 6 to 10 minutes for a job running on a new queue for the first time.
- After a queue is created, if no job is run within one hour, the system releases the queue.

Procedure


1. You can create a queue on the **Overview**, **SQL Editor**, or **Queue Management** page.
 - In the upper right corner of the **Overview** page, click Create Queue.
 - To create a queue on the **Queue Management** page:
 - i. In the navigation pane of the DLI management console, choose **Queue Management**.
 - ii. In the upper right corner of the **Queue Management** page, click Create Queue to create a queue.
 - To create a queue on the **SQL Editor** page:
 - i. On the top menu bar of the DLI management console, click **SQL Editor**.
 - ii. On the left pane of the displayed **SQL Editor** page, click  to the right of **Queues**.
2. In the displayed **Create Queue** dialog box, set related parameters by referring to [Table 6-4](#).

Table 6-4 Parameters

Parameter	Description
Queue	Name of a queue. <ul style="list-style-type: none">• The queue name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_) or be left unspecified.• The length of the name cannot exceed 128 characters.
Queue Type	<ul style="list-style-type: none">• For SQL: compute resources used for SQL jobs.• For general purpose: compute resources used for Spark and Flink jobs. <p>NOTE You can select Dedicated Resource Mode. In this mode, you can create enhanced datasource connections.</p>

Parameter	Description
AZ Mode	The deployment mode for the DLI queue. Select dual-AZ if you require high availability. NOTE <ul style="list-style-type: none">• Currently, only SQL queues support the dual-AZ policy.• Dual-AZ improves data availability by creating a duplicate queue in the second AZ, but at an increased cost (twice as much as that of single AZ mode).• This is a one-time configuration and cannot be changed later.
Queue Specifications Mode	You can select Default or Custom specifications.
CU Specifications	Select queue specifications as required. A CU includes one core and 4 GB memory. <ul style="list-style-type: none">• Fixed specifications include 16 CUs, 64 CUs, 256 CUs, and 512 CUs.• Custom: Set the specifications as required.
Description	Description of the queue to be created. The length of the queue name cannot exceed 256 characters.
Advanced Configuration	<ul style="list-style-type: none">• Default: The system automatically configures the parameter.• Custom CIDR Block: You can specify the CIDR block. For details, see Modifying the Queue CIDR Block. If DLI enhanced datasource connection is used, the CIDR block of the DLI queue cannot overlap with that of the data source. Queue Type: When running an AI-related SQL job, select AI-enhanced. When running other jobs, select Basic.

3. Click **Create Now** to create a queue.

After a queue is created, you can view and select the queue for use on the **Queue Management** page or **SQL Editor** page.

 **NOTE**

It takes 6 to 10 minutes for a job running on a new queue for the first time.

6.4 Deleting a Queue

You can delete a queue based on actual conditions.

 **NOTE**

- This operation will fail if there are jobs in the **Submitting** or **Running** state on this queue.
- The default queue cannot be deleted.

Procedure

Step 1 On the left of the DLI management console, click **Queue Management**.

Step 2 Locate the row where the target queue resides and click **Delete** in the **Operation** column.

If **Delete** in the **Operation** column is gray, the current user does not have the permission of deleting the queue. You can apply to the administrator for the permission.

Step 3 In the dialog box that is displayed, click **OK**.

----End

6.5 Modifying the CIDR Block

If the CIDR block of the DLI queue conflicts with that of the user data source, you can change the CIDR block of the queue.

 **NOTE**

If the queue whose CIDR block is to be modified has jobs that are being submitted or running, or the queue has been bound to enhanced datasource connections, the CIDR block cannot be modified.

Procedure

1. On the left of the DLI management console, click **Queue Management**.
2. Select the queue to be modified and click **Modify CIDR Block** in the **Operation** column.
3. Enter the required CIDR block and click **OK**.

6.6 Modifying Queue Specifications (Manual Scale-out/Scale-in)

Prerequisites

Scaling can be performed for a newly created queue only when there are jobs running on this queue.

Precautions

If **Status of queue xxx is assigning, which is not available** is displayed on the **Modify Queue Specifications** page, the scaling can be performed only after the queue resources are allocated.

Manual Scale-out

If the current queue specifications do not meet service requirements, you can manually change the queue specifications to expand the queue capacity.

NOTE

Scale-out is time-consuming. After you perform scale-out on the **Modify Queue Specifications** page of DLI, wait for about 10 minutes. The duration depends on the value of CU. After a period of time, refresh the **Queue Management** page and check whether the queue size changes to determine whether the scale-out is successful. Alternatively, on the **Job Management** page, check the status of the **SCALE_QUEUE** job. If the job status is **Scaling**, the queue is being scaled out.

Perform the following steps:

1. On the left of the DLI management console, click **Queue Management**.
2. Select the queue to be scaled out and choose **More > Modify Queue Specifications** in the **Operation** column.
3. On the displayed page, select **Scale-out** for **Operation**, set the scale-out amount, and click **OK**.

Manual Scale-in

If the computing service volume is small and the queue specifications are not required, you can manually change the queue specifications to reduce the capacity of the current queue.

NOTE

- Scale-in is time-consuming. After you perform scale-in on the **Modify Queue Specifications** page of DLI, wait for about 10 minutes. The duration is related to the CU value of the scale-in. After a period of time, refresh the **Queue Management** page and check whether the **Queue Specifications** changes to determine whether the scale-in is successful. Alternatively, on the **Job Management** page, check the status of the **SCALE_QUEUE** job. If the job status is **Scaling**, the queue is being scaled in.
- The system may not be fully scaled in to the target size. If the current queue is in use or the service volume of the queue is large, the scale-in may fail or some specifications may be reduced.
- By default, the minimum number of CUs is **16**. That is, when the queue specification is **16 CUs**, scale-in cannot be performed.

Perform the following steps:

1. On the left of the DLI management console, click **Queue Management**.
2. Select the queue to be scaled out and choose **More > Modify Queue Specifications** in the **Operation** column.
3. On the displayed page, select **Scale-in** for **Operation**, set the CU value for scale-in, and click **OK**.

6.7 Scheduling CU Changes (Periodic Scale-out/Scale-in)

Application scenario

Generally, when services are busy, you might need to use more compute resources to process services in a period. After this period, you do not require the same amount of resources. If the purchased queue specifications are small, resources may be insufficient during peak hours. If the queue specification is large, resources may be wasted.

Based on the preceding scenarios, DLI provides the function of periodic task for changing queue specifications. You can set different queue sizes at different time or in different periods based on your service period or usage and the existing queue specifications to meet your service requirements and reduce costs.

Precautions

- Scaling can be performed for a newly created queue only when there are jobs running on this queue.
- A maximum of 12 scheduled tasks can be created for each queue.
- When each scheduled task starts, the actual start time of the specification change has a deviation of 5 minutes. It is recommended that the capacity expansion time be at least 20 minutes earlier than the time when the queue is actually used.
- The interval between two scheduled tasks must be at least 2 hours.
- Changing the specifications of a queue is time-consuming. The time required for changing the specifications depends on the difference between the target specifications and the current specifications. You can view the specifications of the current queue on the **Queue Management** page.
- If a job is running in the current queue, the queue may fail to be scaled in to the target CU value. Instead, it will be scaled in to a value between the current queue specifications and the target specifications. The system will try to scale in again 1 hour later until the next scheduled task starts.
- If a scheduled task does not scale out or scale in to the target CU value, the system triggers the scaling plan again one hour later until the next scheduled task starts.
- The **Task Name**, **After Modification**, and **Executed** parameters of a scheduled task are mandatory. You can set them based on service requirements.
- Scheduled tasks can be scheduled by week in **Repeat**.
 - By default, this parameter is not selected, indicating that the task is executed only once at the time specified by **Executed**.
 - If you select all, the plan is executed every day.
 - If you select some options, the plan is executed once a week at all specified days.

Creating a Periodic Task

- If only scale-out or scale-in is required, you need to create only one scheduled task for changing specifications. Set the **Task Name**, **After Modification**, and **Executed** parameters. For details, see [Table 6-5](#).
- To set both scale-out and scale-in parameters, you need to create two periodic tasks, and set the **Task Name**, **After Modification**, and **Executed** parameters. For details, see [Table 6-5](#).

Perform the following steps:

1. On the left of the DLI management console, click **Queue Management**.
2. Select the queue for which you want to set a periodic task, and choose **More > Schedule CU Changes** in the **Operation** column.
3. On the displayed page, click **Create Periodic Task** in the upper right corner.
4. On the **Create Periodic Task** page, set the required parameters. Click **OK**.

Table 6-5 Parameters

Parameter	Description
Task Name	Enter the name of the periodic task. <ul style="list-style-type: none">• The queue name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_) or be left unspecified.• The length of the database name cannot exceed 128 characters.
Enable Task	Activate the periodic queue scaling task. The default value is Enable . If you set this parameter to Disable , the system does not trigger the periodic task.
Validity Period	Set the time segment for executing the periodic task. The options are Date and Time . If there is no time segment restriction, leave this parameter empty, indicating that the task takes effect permanently. If you need to specify the time segment for the task to take effect, set this parameter based on the service requirements. NOTE <ul style="list-style-type: none">• The start time of the Validity Period must be later than the current system time.• If only scale-out is configured, the system does not automatically scale in after the Validity Period expires. You need to manually modify or configure a periodic scale-in task. That is, a scheduled scaling task is executed at a time.• If both scale-out and scale-in are configured, the system scales in or out resources based on the configured queue specifications within the validity period. After the validity period expires, the system retains the last configured queue specifications.

Parameter	Description
Before Modification	Queue specifications before scale-in or scale-out.
After Modification	Specifications after the queue is scaled in or out. NOTE <ul style="list-style-type: none">• By default, the maximum number of queues is 512 CUs.• By default, the minimum queue specification is 16 CUs. Therefore, when the original specification is 16 CUs, scale-in cannot be performed.• The final specification must be a multiple of 16.
Repeat	Specifies the period for scheduled scaling. NOTE <ul style="list-style-type: none">• You do not need to set this parameter if you only need to perform scale-in or scale-out once.• If you have set scaling, you can set Repeat as required. You can also set the execution period together with the validity period.
Executed	Time when scheduled scale-out or scale-in is performed.

After a periodic task is created, you can view the specification change of the current queue and the latest execution time on the **Schedule CU Changes** page.

Alternatively, on the **Queue Management** page, check whether the **Queue Specifications** change to determine whether the scaling is successful.

Alternatively, on the **Job Management** page, check the status of the **SCALE_QUEUE** job. If the job status is **Scaling**, the queue is being scaled in or out.

Modifying a Periodic Task

If the configured scheduled task does not meet service requirements, you can click **Modify** in the **Operation** column of the periodic task on the Specification Change page to modify the scheduled task.

Deleting a Periodic Task

If you do not need to modify the queue specifications, click **Delete** in the **Operation** column of the scheduled task on the **Schedule CU Specifications** page. In the displayed **Delete Periodic Task** dialog box, click **OK**.

6.8 Testing Address Connectivity

It can be used to test the connectivity between the DLI queue and the peer IP address specified by the user in common scenarios, or the connectivity between

the DLI queue and the peer IP address bound to the datasource connection in cross-source connection scenarios. The operation is as follows:

1. On the **Queue Management** page, locate the row that contains the target queue and choose **More > Test Address Connectivity** in the **Operation** column.
2. On the **Test Address Connectivity** page, enter the address to be tested. The domain name and IP address are supported, and the port number can be specified.
3. Click **Test**.

If the test address is reachable, a message is displayed on the page, indicating that the address is reachable.

If the test address is unreachable, the system displays a message indicating that the address is unreachable. Check the network configurations and try again. Network configurations include the VPC peering and the datasource connection. Check whether they have been activated.

6.9 Creating a Message Notification Topic

Once you have created a message notification topic, you can **Add subscription** of the topic on the **Topic Management** page of the Simple Message Notification service. You can select different ways (such as text messages or emails) to subscribe. After the subscription succeeds, any job failure will automatically be sent to your subscription endpoints. The operation is as follows:

1. On the **Queue Management** page, click **Create SMN Topic** on the upper left side. The **Create SMN Topic** dialog box is displayed.
2. Select a queue and click **OK**. The dialog box is displayed.

NOTE

- You can select a single queue or all queues. If the subscription endpoints of a single queue are different from those of all queues, when a single queue and all queues are both selected, the SMN of all queues do not include the message of the single queue.
 - After a notification topic is created, you will receive a notification only when a session or a batch job fails to be created.
3. Click **Topic Management** to go to the **Topic Management** page of the SMN service.
 4. In the **Operation** column of the corresponding topic in , click **Add Subscription**. Select **Protocol** to determine the subscription mode.
 - If you select **Email**, enter the email address for receiving the confirmation email in the **Endpoint** text box.
 5. After you click the link in the email, you will receive a message indicating that the subscription is successful.
 6. Return to the **Topic Management** page and click **Subscriptions** on the left. If the subscription status is **Confirmed**, the subscription is successful.

7 Data Management

7.1 Databases and Tables

7.1.1 Overview

DLI database and table management provide the following functions:

- [Database Permission Management](#)
- [Table Permission Management](#)
- [Creating a Database or a Table](#)
- [Deleting a Database or a Table](#)
- [Changing the Owners of Databases and Tables](#)
- [Importing Data](#)
- [Exporting Data](#)
- [Viewing Metadata](#)
- [Previewing Data](#)

Difference Between DLI Tables and OBS Tables

- Data stored in DLI tables is applicable to delay-sensitive services, such as interactive queries.
- Data stored in OBS tables is applicable to delay-insensitive services, such as historical data statistics and analysis.

Databases and Tables Page

The **Databases and Tables** page displays all created databases. You can view the database information, such as the owner and the number of tables.

Table 7-1 Database and table management parameters

Parameter	Description
Database Name	<ul style="list-style-type: none">• The database name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_).• The database name is case insensitive and cannot be left unspecified.• It cannot exceed 128 characters.
Owner	Database owner.
Number of Tables	Number of tables in the database.
Description	Description of the database specified during database creation. If no description is provided, -- is displayed.
Operation	<ul style="list-style-type: none">• Manage Permissions: This operation allows you to view the permission information and perform user authorization, permission settings, and user permission revocation.• Manage Tables: This permission allows you to view the tables in the corresponding database. For details, see Table Management Page.• Create Table: This permission allows you to create a table in the corresponding database.• Drop Database: This permission allows you to delete the selected database.• Modify Database. This permission allows you to change the owner of the database. The username must exist under the same account.

Table Management Page

From the **Data Management** page, click the database name or **Manage Tables** in the **Operation** column to switch to the **Table Management** page.

The **Table Management** page displays all tables created in the current database. You can view the table type, data storage location, and other information. Tables are listed in chronological order by default, with the most recently created tables displayed at the top.

Table 7-2 Table management parameters

Parameter	Description
Table Name	<ul style="list-style-type: none">• The table name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_).• The table name is case insensitive and cannot be left unspecified.• The table name can contain the dollar sign (\$). An example value is \$test.• It cannot exceed 128 characters.
Table Type	Table type. Available options are as follows: <ul style="list-style-type: none">• Managed: Indicates that data is stored in a DLI table.• External: Indicates that data is stored in an OBS table.• View: Indicates the view type. You can only create views using SQL statements. NOTE The table or view information contained in the view cannot be modified. If the table or view information is modified, the query may fail.
Owner	User who creates the table.
Storage Location	The location includes DLI, OBS, View, CloudTable, and CSS.
Size	Size of the data in the table. The value is displayed only for tables of the Managed type. For tables of other types, -- is displayed.
Data Source Path	<ul style="list-style-type: none">• If Data Location is OBS, the corresponding OBS path is displayed.• If Data Location is DLI and View, -- is displayed.• When the data storage location is a datasource connection service such as CloudTable and CSS, the corresponding URL is displayed.
Created	Time when the table is created.
Last Accessed	Last time when an operation is performed on the table.

Parameter	Description
Operation	<ul style="list-style-type: none">• Manage Permissions: This operation allows you to view the permission information and perform user authorization, permission settings, and user permission revocation.• Drop: This permission allows you to delete tables from the corresponding database.• More:<ul style="list-style-type: none">- Modify Owner: This operation allows you to change the owner of a table. The username must exist under the same account.- Import: This operation allows you to import data stored in an OBS bucket to a DLI or OBS table.- Export: This operation allows you to export data from a DLI table to an OBS bucket.- Properties: This operation allows you to view data in Metadata and Preview tables.

7.1.2 Database Permission Management

Operation Scenarios

- You can isolate databases allocated to different users by setting permissions to ensure data query performance.
- The administrator and database owner have all permissions, which cannot be set or modified by other users.

Precautions

- Lower-level objects automatically inherit permissions granted to upper-level objects. The hierarchical relationship is database > table > column.
- The database owner, table owner, and **authorized** users can assign permissions on the database and tables.
- Columns can only inherit the query permission. For details about **Inheritable Permissions**, see [Database Permission Management](#).
- The permissions can be revoked only at the initial level to which the permissions are granted. You need to grant and revoke permissions at the same level. For example, after you are granted the insertion permission on a database, you can obtain the insertion permission on the tables in the database. Your insertion permission can be revoked only at the database level.

Viewing Database Permissions

1. On the left of the management console, choose **Data Management > Databases and Tables**.
2. Locate the row where the target database resides and click **Manage Permissions** in the **Operation** column.

Permissions can be granted to new users or projects, modified for users or projects with existing permissions, or revoked from a user or project.

Granting Permissions to a New User or Project

Here, the new user or project refers to a user or a project that does not have permissions on the database.

1. Click a database you need. In the displayed **Database Permission Management** page, click **Grant Permission** in the upper right corner.
2. In the displayed dialog box, select **User** or **Project**, enter the username or select the project to be authorized, and select the required permissions. For details about the permissions, see [Table 7-3](#).

Table 7-3 Parameters

Parameter	Description
Authorization Object	Select User or Project .
Username/ Project Name	<ul style="list-style-type: none">• If you select User, enter the IAM username when adding a user to the database. NOTE The username is an existing IAM user name and has logged in to the DLI management console.• If you select Project, select the project to be authorized in the current region. NOTE When Project is selected:<ul style="list-style-type: none">• If you set Non-inheritable Permissions, you cannot view tables in the corresponding database within the project.• If you set Inheritable Permissions, you can view all tables in the database within the project.

Parameter	Description
Non-inheritable Permissions	<p>Select a permission to grant it to the user, or deselect a permission to revoke it.</p> <p>Non-inheritable permissions apply only to the current database.</p> <ul style="list-style-type: none">• The following permissions are applicable to both user and project authorization:<ul style="list-style-type: none">- Drop Database: This permission allows you to delete the current database.- Create Table: This permission allows you to create tables in the current database.- Create View: This permission allows you to create views in the current database.- Execute SQL EXPLAIN: This permission allows you to execute an EXPLAIN statement and view information about how this database executes a query.- Create Role: This permission allows you to create roles in the current database.- Delete Role: This permission allows you to delete roles of the current database.- View Role: This permission allows you to view the role of the current user.- Bind Role: This permission allows you to bind roles to the current database.- Unbind Role: This permission allows you to bind roles from the current database.- View All Binding Relationships: This permission allows you to view the binding relationships between all roles and users.- Create Function: This permission allows you to create a function in the current database.- Delete Function: This permission allows you to delete functions from the current database.- View All Functions: This permission allows you to view all functions in the current database.- View Function Details: This permission allows you to view details about the current function.• The following permissions can only be granted to users:<ul style="list-style-type: none">- View All Tables: This permission allows you to view all tables in the current database.- View Database: This permission allows you to view the information about the current database.

Parameter	Description
Inheritable Permissions	<p>Select a permission to grant it to the user, or deselect a permission to revoke it.</p> <p>Inheritable permissions are applicable to the current database and all its tables. However, only the query permission is applicable to table columns.</p> <p>The following permissions can be granted to both user and project.</p> <ul style="list-style-type: none">● Drop Table: This permission allows you to delete tables in a database.● Select Table: This permission allows you to query data of the current table.● View Table Information: This permission allows you to view information about the current table.● Insert: This permission allows you to insert data into the current table.● Add Column: This permission allows you to add columns to the current table.● Overwrite: This permission allows you to insert data to overwrite the data in the current table.● Grant Permission: This permission allows you to grant database permissions to other users or projects.● Revoke Permission: This permission allows you to revoke the permissions of the database that other users have but cannot revoke the database owner's permissions.● Add Partition to Partition Table: This permission allows you to add a partition to a partition table.● Delete Partition from Partition Table: This permission allows you to delete existing partitions from a partition table.● Configure Path for Partition: This permission allows you to set the path of a partition in a partition table to a specified OBS path.● Rename Table Partition: This permission allows you to rename partitions in a partition table.● Rename Table: This permission allows you to rename tables.● Restore Table Partition: This permission allows you to export partition information from the file system and save the information to metadata.● View All Partitions: This permission allows you to view all partitions in a partition table.● View Other Users' Permissions: This permission allows you to query other users' permission on the current database.

3. Click **OK**.

Modifying Permissions for an Existing User or Project

For a user or project that has some permissions on the database, you can revoke the existing permissions or grant new ones.

NOTE

If the options in **Set Permission** are gray, the corresponding account does not have the permission to modify the database. You can apply to the administrator, database owner, or other authorized users for granting and revoking permissions of databases.

1. In the **User Permission Info** list, find the user whose permission needs to be set.
 - If the user is a sub-user, you can set permissions for it.
 - If the user is already an administrator, you can only view the permissions information.

In the **Project Permission Info** list, locate the project for which you want to set permissions and click **Set Permission**.

2. In the **Operation** column of the sub-user or project, click **Set Permission**. The **Set Permission** dialog box is displayed.

For details about the permissions of database users or projects, see [Table 7-3](#).

3. Click **OK**.

Revoking All Permissions of a User or Project

Revoke all permissions of a user or a project.

- In the user list under **User Permission Info**, locate the row where the target sub-user resides and click **Revoke Permission** in the **Operation** column. In the displayed dialog box, click **OK**. In this case, the user has no permissions on the database.

NOTE

If a user is an administrator, **Revoke Permission** is gray, indicating that the user's permission cannot be revoked.

- In the **Project Permission Info** area, select the project whose permissions need to be revoked and click **Revoke Permission** in the **Operation** column. After you click **OK**, the project does not have any permissions on the database.

7.1.3 Table Permission Management

Operation Scenario

- You can isolate databases allocated to different users by setting permissions to ensure data query performance.
- The administrator and database owner have all permissions, which cannot be set or modified by other users.

- When setting database permissions for a new user, ensure that the user group to which the user belongs has the **Tenant Guest** permission.

Viewing Table Permissions

1. On the left of the management console, choose **Data Management > Databases and Tables**.
2. Click the database name in the table whose authority is to be set. The **Table Management** page of the database is displayed.
3. Locate the row where the target table resides and click **Manage Permissions** in the **Operation** column.

Permissions can be granted to new users or projects, modified for users or projects with existing permissions, or revoked from a user or project.

Granting Permissions to a New User or a Project

Here, the new user or project refers to a user or a project that does not have permissions on the database.

1. Click the table you need. In the displayed table permissions page, click **Grant Permission** in the upper right corner.
2. In the displayed **Grant Permission** dialog box, select the required permissions.
 - For details about the DLI table permissions, see [Table 7-4](#).

Table 7-4 Parameter description

Parameter	Description
Authorization Object	Select User or Project .
Username/Project Name	<ul style="list-style-type: none">• If you select User, enter the IAM username when granting table permissions to the user. NOTE The username is an existing IAM user name and has logged in to the DLI management console.• If you select Project, select the project to be authorized in the current region. NOTE If you select Project, you can only view information about the authorized tables and their databases.

Parameter	Description
Non-inheritable Permissions	<p>Select a permission to grant it to the user, or deselect a permission to revoke it.</p> <ul style="list-style-type: none"> ● The following permissions are applicable to both user and project authorization: <ul style="list-style-type: none"> – Select Table: This permission allows you to query data of the current table. – View Table Information: This permission allows you to view information about the current table. – View Table Creation Statement: This permission allows you to view the statement for creating the current table. – Drop Table: This permission allows you to delete the current table. – Rename Table: Rename the current table. – Insert: This permission allows you to insert data into the current table. – Overwrite: This permission allows you to insert data to overwrite the data in the current table. – Add Column: Add columns to the current table. – Grant Permission: The current user can grant table permissions to other users. – Revoke Permission: The current user can revoke the table's permissions that other users have but cannot revoke the table owner's permissions. – View Other Users' Permissions: This permission allows you to query other users' permission on the current table. <p>The partition table also has the following permissions:</p> <ul style="list-style-type: none"> – Delete Partition: This permission allows you to delete existing partitions from a partition table. – View All Partitions: This permission allows you to view all partitions in a partition table. <ul style="list-style-type: none"> ● The following permissions can only be granted to users: <ul style="list-style-type: none"> – View Table: This permission allows you to display the current table.
Column Permissions	<ul style="list-style-type: none"> ● Only the query permission on table columns can be inherited. If you select a column name, you have the permission to query the corresponding column. ● If Query Table is selected in Non-inherited Permissions, the Column Permissions is automatically hidden.

- For details about the OBS table permissions, see [Table 7-5](#).

Table 7-5 Parameter description

Parameter	Description
Authorization Object	Select User or Project .
Username/ Project Name	<ul style="list-style-type: none"> • If you select User, enter the IAM username when granting table permissions to the user. NOTE The username is an existing IAM user name and has logged in to the DLI management console. • If you select Project, select the project to be authorized in the current region. NOTE If you select Project, you can only view information about the authorized tables and their databases.

Parameter	Description
Non-inheritable Permissions	<p>Select a permission to grant it to the user, or deselect a permission to revoke it.</p> <ul style="list-style-type: none"> • The following permissions are applicable to both user and project authorization: <ul style="list-style-type: none"> - View Table Creation Statement: This permission allows you to view the statement for creating the current table. - View Table Information: This permission allows you to view information about the current table. - Select Table: This permission allows you to query data of the current table. - Drop Table: This permission allows you to delete the current table. - Rename Table: Rename the current table. - Insert: This permission allows you to insert data into the current table. - Overwrite: This permission allows you to insert data to overwrite the data in the current table. - Add Column: This permission allows you to add columns to the current table. - Grant Permission: This permission allows you to grant table permissions to other users or projects. - Revoke Permission: This permission allows you to revoke the table's permissions that other users or projects have but cannot revoke the table owner's permissions. - View Other Users' Permissions: This permission allows you to query other users' permission on the current table. <p>The partition table also has the following permissions:</p> <ul style="list-style-type: none"> - Add Partition: This permission allows you to add a partition to a partition table. - Delete Partition: This permission allows you to delete existing partitions from a partition table. - Configure Path for Partition: This permission allows you to set the path of a partition in a partition table to a specified OBS path. - Rename Table Partition: This permission allows you to rename partitions in a partition table. - Restore Table Partition: This permission allows you to export partition information from the file system and save the information to metadata. - View All Partitions: This permission allows you to view all partitions in a partition table.

Parameter	Description
	<ul style="list-style-type: none"> The following permissions can only be granted to users: <ul style="list-style-type: none"> View Table: This permission allows you to view the current table.
Column permissions	<ul style="list-style-type: none"> Only the query permission on table columns can be inherited. If you select a column name, you have the permission to query the corresponding column. If Query Table is selected in Non-inherited Permissions, the Column Permissions is automatically hidden.

- For details about the view permissions, see [Table 7-6](#).

 **NOTE**

A view can be created only by using SQL statements. You cannot create a view on the **Create Table** page.

Table 7-6 Parameter description

Parameter	Description
Authorization Object	Select User or Project .
Username/Project Name	<ul style="list-style-type: none"> If you select User, enter the IAM username when adding a user to the database. <p>NOTE The username is an existing IAM user name and has logged in to the DLI management console.</p> If you select Project, select the project to be authorized in the current region. <p>NOTE If you select Project, you can only view information about the authorized tables and their databases.</p>

Parameter	Description
Non-inheritable Permissions	<p>Select a permission to grant it to the user, or deselect a permission to revoke it.</p> <ul style="list-style-type: none"> • The following permissions are applicable to both user and project authorization: <ul style="list-style-type: none"> - View Table Information: This permission allows you to view information about the current table. - View Table Creation Statement: This permission allows you to view the statement for creating the current table. - Drop Table: This permission allows you to delete the current table. - Select Table: This permission allows you to query data of the current table. - Rename Table: Rename the current table. - Grant Permission: The current user or project can grant table permissions to other users or projects. - Revoke Permission: The current user or project can revoke the table's permissions that other users or projects have but cannot revoke the table owner's permissions. - View Other Users' Permissions: This permission allows you to query other users' permission on the current table. • Only applicable to <ul style="list-style-type: none"> - View Table: This permission allows you to view the current table.

3. Click **OK**.

Modifying Permissions for an Existing User or Project

For a user or project that has some permissions on the database, you can revoke the existing permissions or grant new ones.

NOTE

If all options under **Set Permission** are gray, you are not allowed to change permissions on this table. You can apply to the administrator, table owner, or other authorized users for granting and revoking table permissions.

1. In the **User Permission Info** list, find the user whose permission needs to be set.
 - If the user is a sub-user and is not the owner of the table, you can set permissions.
 - If the user is an administrator or table owner, you can only view permissions.

- In the **Project Permission Info** list, locate the project for which you want to set permissions and click **Set Permission**.
- In the **Operation** column of the sub-user or project, click **Set Permission**. The **Set Permission** dialog box is displayed.
 - For details about DLI table user or project permissions, see [Table 7-4](#).
 - For details about OBS table user or project permissions, see [Table 7-5](#).
 - For details about View table user or project permissions, see [Table 7-6](#).
 - Click **OK**.

Revoking All Permissions of a User or Project

Revoke all permissions of a user or a project.

- In the user list under **User Permission Info**, locate the row where the target sub-user resides and click **Revoke Permission** in the **Operation** column. In the displayed dialog box, click **OK**. In this case, the user has no permissions on the table.

NOTE

In the following cases, **Revoke Permission** is gray, indicating that the permission of the user cannot be revoked.

- The user is an administrator.
 - The sub-user is the owner of the table.
 - The sub-user has only inheritable permissions.
- In the **Project Permission Info** area, select the project whose permissions need to be revoked and click **Revoke Permission** in the **Operation** column. After you click **OK**, the project does not have any permissions on the table.

NOTE

If a project has only inheritable permissions, **Revoke Permission** is gray, indicating that the permissions of the project cannot be revoked.

7.1.4 Creating a Database or a Table

Definition of Database and Table in DLI

A database, built on the computer storage device, is a data warehouse where data is organized, stored, and managed based on its structure.

The table is an important part of the database. It consists of rows and columns. Each column functions as a field. Each value in a field (column) represents a type of data.

The database is a framework and the table contains data content. A database has one or more tables.

The basic database concept and usage in DLI are similar to those in the Oracle database. DLI permissions are granted on a per database basis.

You can create databases and tables on the management console or using SQL statements. This section describes how to create a database and a table on the management console.

 NOTE


A view can be created only by using SQL statements. You cannot create a view on the **Create Table** page.

Precautions

- If a folder and a file have the same name in the OBS directory, the file path is preferred as the path of the OBS table to be created.

Creating a Database

Step 1 You can create a database on either the **Data Management** page or the **SQL Editor** page.

- To create a database on the **Data Management** page:
 - a. On the left of the management console, choose **Data Management > Databases and Tables**.
 - b. In the upper right corner of the **Databases and Tables** page, click **Create** to create a database.
- To create a database on the **SQL Editor** page:
 - a. On the left of the management console, click **SQL Editor**.
 - b. In the navigation pane on the left, click  beside **Databases**, you can create a database.

Step 2 In the displayed **Create Database** dialog box, specify **Name** and **Description** by referring to [Table 7-7](#).

Table 7-7 Parameter description

Parameter	Description	Example
Database Name	<ul style="list-style-type: none">• The database name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_).• The database name is case insensitive and cannot be left blank.• The length of the database name cannot exceed 128 characters.	DB01
Description	Description of a database.	-

 NOTE

The **default** database is a built-in database. You cannot create the database named **default**.

Step 3 Click **OK**.

After a database is created, you can view and select the database for use on the **Databases and Tables** page or **SQL Editor** page.



----End

Creating a Table

Before creating a table, ensure that a database has been created.

Step 1 You can create a table on either the **Databases and Tables** page or the **SQL Editor** page. **NOTE**

Datasource connection tables, such as View tables, HBase (CloudTable/MRS) tables, OpenTSDB (CloudTable/MRS) tables, DWS tables, RDS tables, and CSS tables, cannot be created. You can use SQL to create views and datasource connection tables. For details, see sections **Creating a View** and **Creating a Datasource Connection Table** in the *Data Lake Insight SQL Syntax Reference*.

- To create a table on the **Data Management** page:
 - a. On the left of the management console, choose **Data Management > Databases and Tables**.
 - b. On the **Databases and Tables** page, select the database for which you want to create a table. Click **Create Table** in the **Operation** column to create a table in the current database.
- To create a table on the **SQL Editor** page:
 - a. On the left of the management console, click **SQL Editor**.
 - b. In the navigation pane of the displayed **SQL Editor** page, click **Databases**. You can create a table in either of the following ways:
 - Click a database name. In the **Tables** area, click  on the right to create a table in the current database.
 - Click  on the right of the database and choose **Create Table** from the shortcut menu to create a table in the current database.

Step 2 In the displayed **Create Table** dialog box, set parameters as required.

- If you set **Data Location** to **DLI**, set related parameters by referring to [Table 7-8](#).
- If you set **Data Location** to **OBS**, set related parameters by referring to [Table 7-8](#) and [Table 7-9](#).

Table 7-8 Common parameters

Parameter	Description	Example
Name	<ul style="list-style-type: none">- The table name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_).- The table name is case insensitive and cannot be left unspecified.- The table name can contain the dollar sign (\$). An example value is \$test.- The length of the table name cannot exceed 128 characters.	table01
Data Location	Data storage location. Currently, DLI and OBS are supported.	OBS
Table Description	Description of the table.	-
Column Name	<p>Name of a column in a table. The column name must contain at least one letter. It can contain underscores (_), but cannot contain only digits or the combination of digits and letters B, K, M, G, and D (with the letter as the end).</p> <p>You can select Normal or Partition. Partition columns are dedicated to partition tables. User data is partitioned to improve query efficiency.</p> <p>NOTE The column name is case-insensitive and must be unique.</p>	name



Parameter	Description	Example
Type	<p>Data type of a column. This parameter corresponds to Column Name.</p> <ul style="list-style-type: none"> - int: The storage space is four bytes. - string: Refers to the character string type. - float: Refers to the single-precision floating point. The storage space is four bytes. - double: The storage space is eight bytes. - date: The date ranges from 0000-01-01 to 9999-12-31. - boolean: The storage space is one byte. - decimal: The valid bits are positive integers between 1 to 38, including 1 and 38. The decimal digits are integers less than 10. - smallint/short: The storage space is two bytes. - bigint/long: The storage space is eight bytes. - tinyint: The storage space is one byte. Only OBS tables support this function. - timestamp: Refers to the date and time, which can be accurate to six decimal points. - char: Refers to the fixed-length character strings. - varchar: Refers to the variable-length strings. 	string
Column Description	Description of a column.	-
Operation	<ul style="list-style-type: none"> -  : Add Column -  : Delete Column <p>NOTE If the table to be created includes a great number of columns, you are advised to use SQL statements to create the table or import column information from the local EXCEL file.</p>	-

Table 7-9 Parameter description when **Data Location** is set to **OBS**

Parameter	Description	Example
Data Format	<p>DLI supports the following data formats:</p> <ul style="list-style-type: none"> - Parquet: DLI can read non-compressed data or data that is compressed using Snappy and gzip. - CSV: DLI can read non-compressed data or data that is compressed using gzip. - ORC: DLI can read non-compressed data or data that is compressed using Snappy. - JSON: DLI can read non-compressed data or data that is compressed using gzip. - Carbon: DLI can read non-compressed carbon data. - Avro: DLI can read uncompressed Avro data. 	CSV
Bucket Location	<p>Enter or select an OBS path. The path can be a folder or a path.</p> <p>NOTE If you need to import data stored in OBS to the OBS table, set this parameter to the path of a folder. If the table creation path is a file, data fails to be imported.</p>	obs://obs1/sampleddata.csv
Table Header: No/Yes	<p>This parameter is valid only when Data Format is set to CSV. Whether the data source to be imported contains the table header.</p> <p>Click Advanced Settings and select the check box next to Table Header: No. If the check box is selected, the table header is displayed. If the check box is deselected, no table header is displayed.</p>	-

Parameter	Description	Example
User-defined Delimiter	This parameter is valid only when Data Format is set to CSV and you select User-defined Delimiter . The following delimiters are supported: <ul style="list-style-type: none"> - Comma (,) - Vertical bar () - Tab character (\t) - Others: Enter a user-defined delimiter. 	Comma (,)
User-defined Quotation Character	This parameter is valid only when Data Format is set to CSV and you select User-defined Quotation Character . The following quotation characters are supported: <ul style="list-style-type: none"> - Single quotation mark (') - Double quotation marks (") - Others: Enter a user-defined quotation character. 	Single quotation mark (')
User-defined Escape Character	This parameter is valid only when Data Format is set to CSV and you select User-defined Escape Character . The following escape characters are supported: <ul style="list-style-type: none"> - Backslash (\) - Others: Enter a user-defined escape character. 	Backslash (\)
Date Format	This parameter is valid only when Data Format is set to CSV or JSON . This parameter specifies the format of the date in the table and is valid only Advanced Settings is selected. The default value is yyyy-MM-dd .	2000-01-01
Timestamp Format	This parameter is valid only when Data Format is set to CSV or JSON . This parameter specifies the format of the timestamp in the table and is valid only Advanced Settings is selected. The default value is yyyy-MM-dd HH:mm:ss .	2000-01-01 09:00:00

Step 3 Click **OK**.

After a table is created, you can view and select the table for use on the **Data Management** page or **SQL Editor** page.

Step 4 (Optional) After a DLI table is created, you can decide whether to directly import data to the table.

----End

7.1.5 Deleting a Database or a Table

You can delete unnecessary databases and tables based on actual conditions.

Precautions


- You are not allowed to delete databases or tables that are being used for running jobs.
- The administrator, database owner, and users with the database deletion permission can delete the database. The administrator, database owner, and users with the table deletion permission can delete the table.

NOTE

If a database or table is deleted, it cannot be recovered. Exercise caution when performing this operation.

Deleting a Table

You can delete a table on either the **Data Management** page or the **SQL Editor** page.

- Delete the table on the **Data Management** page.
 - a. On the left of the management console, choose **Data Management > Databases and Tables**.
 - b. Locate the row where the database whose tables you want to delete, click the database name to switch to the **Table Management** page.
 - c. Locate the row where the target table resides and click **More > Drop** in the **Operation** column.
 - d. In the dialog box that is displayed, click **OK**.
- Delete a table on the **SQL Editor** page.
 - a. On the top menu bar of the DLI management console, click **SQL Editor**.
 - b. In the navigation tree on the left, click **Databases**. Click the name of the database whose table is to be deleted. The tables in the selected database are displayed.
 - c. Click  on the right of the table and choose **Drop Table** from the shortcut menu.
 - d. In the dialog box that is displayed, click **OK**.

Deleting a Database

1. On the left of the management console, choose **Data Management > Databases and Tables**.

2. Locate the row where the target database resides and click **More > Drop Database** in the **Operation** column.

 **NOTE**

You cannot delete databases that contain tables. To delete a database containing tables, delete the tables first.


3. In the dialog box that is displayed, click **OK**.

7.1.6 Modifying the Owners of Databases and Tables

During actual use, developers create databases and tables and submit them to test personnel for testing. After the test is complete, the databases and tables are transferred to O&M personnel for user experience. In this case, you can change the owner of the databases and tables to transfer data to other owners.

Modifying the Database Owner

You can change the owner of a database on either the **Data Management** page or the **SQL Editor** page.

- On the **Data Management** page, change the database owner.
 - a. On the left of the management console, choose **Data Management > Databases and Tables**.
 - b. On the **Databases and Tables** page, select the database to be modified and click **Modify** in the **Operation** column.
 - c. In the displayed dialog box, enter a new owner name (an existing username) and click **OK**.
- Change the database owner on the **SQL Editor** page.
 - a. On the left of the management console, click **SQL Editor**.
 - b. In the navigation tree on the left, click **Databases and Tables**, click  on the right of the corresponding database, and choose **Modify Database** from the shortcut menu.
 - c. In the displayed dialog box, enter a new owner name (an existing username) and click **OK**.

Modifying the Table Owner

1. On the left of the management console, choose **Data Management > Databases and Tables**.
2. Click the name of the database corresponding to the table to be modified. The **Manage Tables** page of the database is displayed.
3. In the **Operation** column of the target table, choose **More > Modify Owner**.
4. In the displayed dialog box, enter a new owner name (an existing username) and click **OK**.

7.1.7 Importing Data to the Table

You can import data from OBS to a table created in DLI.


Precautions

- When creating an OBS table, you must specify a folder as the directory. If a file is specified, data import may be failed.
- Only one path can be specified during data import. The path cannot contain commas (,).
- If a folder and a file have the same name in the OBS directory, the file path is preferred as the path of the data to be imported.
- To import data in CSV format to a partitioned table, place the column to be partitioned in the last column of the data source.
- You are advised not to concurrently import data in to a table. If you concurrently import data into a table, there is a possibility that conflicts occur, leading to failed data import.
- The imported file can be in CSV, Parquet, ORC, JSON, and Avro format. The text format must be UTF-8.

Prerequisites

The data to be imported has been stored on OBS.


Procedure

- Step 1** You can import data on either the **Data Management** page or the **SQL Editor** page.
- To import data on the **Data Management** page:
 - a. On the left of the management console, choose **Data Management > Databases and Tables**.
 - b. Click the name of the database corresponding to the table where data is to be imported to switch to the **Manage Tables** page.
 - c. Locate the row where the target table resides and choose **More > Import** in the **Operation** column to switch to the **Import Data** dialog box.
 - To import data on the **SQL Editor** page:
 - a. On the left of the management console, click **SQL Editor**.
 - b. In the navigation tree on the left of **SQL Editor**, click **Databases** to see all databases. Click the database name corresponding to the table to which data is to be imported. The tables are displayed.
 - c. Click  on the right of the table and choose **Import** from the shortcut menu. The **Import Data** page is displayed.

- Step 2** On the **Import Data** page, set the parameters based on [Table 7-10](#).

Table 7-10 Description

Parameter	Description	Example
Database Name	Database where the current table is located.	-
Table Name	Name of the current table.	-

Parameter	Description	Example
File Format	Format of the data source file to be imported. The CSV, Parquet, ORC, JSON, and Avro formats are supported.	CSV
Queue	Select a queue.	-
Path	You can directly enter a path or click  and select an OBS path. If no bucket is available, you can directly switch to the OBS management console and create an OBS bucket. NOTE The path can be a file or folder.	obs://DLI/ sampledat a.csv
Table Header: No/Yes	This parameter is valid only when File Format is set to CSV . Whether the data source to be imported contains the table header. Click Advanced Settings and select the check box next to Table Header: No . If the check box is selected, the table header is displayed. If the check box is deselected, no table header is displayed.	-
User-defined Delimiter	This parameter is valid only when File Format is set to CSV and you select User-defined Delimiter . The following delimiters are supported: <ul style="list-style-type: none"> • Comma (,) • Vertical bar () • Tab character (\t) • Others: Enter a user-defined delimiter. 	Default value: comma (,)
User-defined Quotation Character	This parameter is valid only when File Format is set to CSV and User-defined Quotation Character is selected. The following quotation characters are supported: <ul style="list-style-type: none"> • Single quotation mark (') • Double quotation marks (") • Others: Enter a user-defined quotation character. 	Default value: double quotation marks (")
User-defined Escape Character	This parameter is valid only when File Format is set to CSV and you select User-defined Escape Character . The following escape characters are supported: <ul style="list-style-type: none"> • Backslash (\) • Others: Enter a user-defined escape character. 	Default value: backslash (\)


Parameter	Description	Example
Date Format	This parameter is valid only when File Format is set to CSV or JSON . This parameter specifies the format of the date in the table and is valid only Advanced Settings is selected. The default value is yyyy-MM-dd .	2000-01-01
Timestamp Format	This parameter is valid only when File Format is set to CSV or JSON . This parameter specifies the format of the timestamp in the table and is valid only Advanced Settings is selected. The default value is yyyy-MM-dd HH:mm:ss .	2000-01-01 09:00:00
Error Records Path	This parameter is valid only when File Format is set to CSV or JSON . The parameter specifies the error data is stored in the corresponding OBS path and is valid only Advanced Settings is selected.	obs://DLI/

Step 3 Click **OK**.

Step 4 You can view the imported data in either of the following ways:

 **NOTE**

Currently, only the first 10 records are displayed.

- Choose **Data Management > Databases and Tables > Manage Tables**, locate the row that contains the target table, click **More** in the **Operation** column, and select **Properties**. In the displayed **Preview** tab, you can view the imported data.
- On the **Databases** tab page of the **SQL Editor**, click the database name to go to the table list. Click  on the right of the table and choose **Properties** from the shortcut menu. In the displayed dialog box, click **Preview** to view the imported data.

Step 5 (Optional) View the status and execution result of the importing job on the **Job Management** page.

----End

7.1.8 Exporting Data from DLI to OBS

You can export data from a DLI table to OBS. During the export, a folder is created in OBS or the content in the existing folder is overwritten.

Precautions

- The exported file can be in JSON format, and the text format can only be UTF-8.
- Only the data in the DLI table (the table type is **Managed**) can be exported to the OBS bucket, and the export path must be specified to the folder level.

- Data can be exported across accounts. That is, after account B authorizes account A, account A can export data to the OBS path of account B if account A has the permission to read the metadata and permission information about the OBS bucket of account B and read and write the path.

Procedure


- Step 1** You can export data on either the **Data Management** page or the **SQL Editor** page.
- To import data on the **Data Management** page:
 - a. On the left of the management console, choose **Data Management > Databases and Tables**.
 - b. Click the name of the database corresponding to the table where data is to be exported to switch to the **Manage Tables** page.
 - c. Select the corresponding table (DLI table) and choose **More > Export** in the **Operation** column. The **Export Data** page is displayed.
 - To export data on the **SQL Editor** page:
 - a. On the left of the management console, click **SQL Editor**.
 - b. In the navigation tree on the left, click **Databases** to see all databases. Click the database name corresponding to the table to which data is to be exported. The tables are displayed.
 - c. Click  on the right of the managed table (DLI table) whose data is to be exported, and choose **Export** from the shortcut menu.
- Step 2** In the displayed **Export Data** dialog box, specify parameters by referring to [Table 7-11](#).


Table 7-11 Parameter description

Parameter	Description
Database Name	Database where the current table is located.
Table Name	Name of the current table.
Export Format	Format of the file storing data to be exported. Formats other than JSON will be supported in later versions.
Queue	Select a queue.
Compression Format	Compression format of the data to be exported. The following compression formats are supported: <ul style="list-style-type: none">• none• bzip2• deflate• gzip

Parameter	Description
Export Path	<ul style="list-style-type: none">• Enter or select an OBS path.• The export path must be a folder that does not exist in the OBS bucket. Specifically, you need to create a folder in the target OBS directory.• The folder name cannot contain the special characters of \/:*?"<> , and cannot start or end with a dot (.).
Export Mode	Storage mode of the data to be exported. <ul style="list-style-type: none">• New OBS directory: If the specified export directory exists, an error is reported and the export operation cannot be performed.• Existing OBS directory (Overwritten): If you create a file in the specified directory, the existing file will be overwritten.
Table Header: No/Yes	This parameter is invalid only when Export Format is set to json . Formats other than JSON will be supported in later versions. Whether the data to be exported contains the table header. Click Advanced Settings and select the check box next to Table Header: No . If the check box is selected, the table header is displayed. If the check box is deselected, no table header is displayed.

Step 3 Click **OK**.

Step 4 (Optional) You can view the job status (indicated by **Status**), statements (indicated by **Statement**), and other information about exporting jobs on the **SQL Jobs** page.

1. Select **EXPORT** from the **Job Type** drop-down list box and specify the time range for exporting data. The jobs meeting the requirements are displayed in the job list.
2. Click  to view details about an exporting job.

----End


7.1.9 Viewing Metadata

Metadata Description

- Metadata is used to define data types. It describes information about the data, including the source, size, format, and other data features. In database fields, metadata interprets data content in the data warehouse.
- When you create a table, metadata is defined, consisting of the column name, type, and description.
- The **Metadata** page displays information about the target table, including **Column Name**, **Column Type**, **Data Type**, and **Description**.

Procedure

You can view metadata on either the **Data Management** page or the **SQL Editor** page.

- To view metadata on the **Data Management** page:
 - a. On the left of the management console, choose **Data Management > Databases and Tables**.
 - b. On the displayed **Data Management** page, click the name of the database where the target table whose data you want to export resides to switch to the **Manage Tables** page.
 - c. Click **More** in the **Operation** column of the target table and select **Properties**. On the **Metadata** tab page, view the metadata of the table.
- To view metadata on the **SQL Editor** page:
 - a. On the left of the management console, click **SQL Editor**.
 - b. In the navigation pane of the displayed **SQL Editor** page, click **Databases**.
 - c. Click the corresponding database name to view the tables in the database.
 - d. Click  on the right of the table and choose **Properties** from the shortcut menu. On the **Metadata** tab page, view the metadata of the table.

7.1.10 Previewing Data

The **Preview** page displays the first 10 records in the table.

Procedure

You can preview data on either the **Data Management** page or the **SQL Editor** page.

- To preview data on the **Data Management** page:
 - a. On the left of the management console, choose **Data Management > Databases and Tables**.
 - b. On the displayed **Data Management** page, click the name of the database where the target table whose data you want to export resides to switch to the **Manage Tables** page.
 - c. Click **More** in the **Operation** column of the target table and select **Properties**.
 - d. Click the **Preview** tab to preview the table data.
- To preview data on the **SQL Editor** page:
 - a. On the left of the management console, click **SQL Editor**.
 - b. In the navigation pane of the displayed **SQL Editor** page, click **Databases**.
 - c. Click the corresponding database name to view the tables in the database.

- d. Click  on the right of the corresponding table, choose **Properties** from the list menu, and click the **Preview** tab to preview the data of the table.

7.2 Package Management

7.2.1 Overview

Package management provides the following functions:

- [Managing Package Permissions](#)
- [Creating a Package](#)
- [Deleting a Package](#)

 **NOTE**

You can delete program packages in batches.

- [Modifying the Owner](#)

Package Management Page

Table 7-12 Package management parameters

Parameter	Description
Group Name	Name of the group to which the package belongs. If the package is not grouped, - is displayed.
Package Name	Name of a package.
Owner	Name of the user who uploads the package.
Type	Type of a package. Available options are as follows: <ul style="list-style-type: none">• JAR: JAR file• PyFile: User Python file• File: User file• ModelFile: User AI model file
Status	Status of the package to be created. <ul style="list-style-type: none">• Uploading: The file is being uploaded.• Finished: The resource package has been uploaded.• Failed: The resource package upload failed.
Created	Time when a package is created.
Updated	Time when the package is updated.
Operation	Manage Permissions: Manage user permissions for a package. Modify Owner: Modify the owner of the package. Delete: Delete the package.

7.2.2 Managing Permissions on Packages and Package Groups

Scenarios

- You can isolate package groups or packages allocated to different users by setting permissions to ensure data query performance.
- The administrator and the owner of a package group or package have all permissions. You do not need to set permissions and the permissions cannot be modified by other users.
- When you set permissions on a package group or a package to a new user, the user group to which the user belongs must have the Tenant Guest permission.

On the **Package Management** page, click **Manage Permissions** in the **Operation** column of the target package. On the displayed **User Permission Info** page, you can grant permissions for the package group or package, set and revoke user permissions.

NOTE

- If you select a group when creating a package, you can manage permissions of the corresponding program package group.
- If you select **No grouping** when creating a package, you can manage permissions of the corresponding package.

Granting Permissions on Package Groups/Packages

Click **Grant Permission** in the upper right corner of the page.

- Granting permissions on package groups

Table 7-13 Permission parameters

Parameter	Description
Username	Name of the authorized IAM user. NOTE The username is the name of an existing IAM user.

Parameter	Description
Select the permissions to be granted to the user	<ul style="list-style-type: none"> • Use Group: This permission allows you to use the package of this group. • Update Group: This permission allows you to update the packages in the group, including creating a package in the group. • Query Group: This permission allows you to query the details of a package in a group. • Delete Group: This permission allows you to delete the package of the group. • Grant Permission: This permission allows you to grant group permissions to other users. • Revoke Permission: This permission allows you to revoke the group permissions that other users have but cannot revoke the group owner's permissions. • View Other User's Permissions: This permission allows you to view the group permissions of other users.

- Granting permissions on packages

Table 7-14 Permission parameters

Parameter	Description
Username	Name of the authorized IAM user. NOTE The username is the name of an existing IAM user.
Select the permissions to be granted to the user	<ul style="list-style-type: none"> • Use Package: This permission allows you to use the package. • Update Package: This permission allows you to update the package. • Query Package: This permission allows you to query the package. • Delete Package: This permission allows you to delete the package. • Grant Permission: This permission allows you to grant package permissions to other users. • Revoke Permission: This permission allows you to revoke the package permissions that other users have but cannot revoke the package owner's permissions. • View Other User's Permissions: This permission allows you to view the package permissions of other users.

Setting Permissions on Package Groups and Packages

Click **Set Permission** in the **Operation** column of the sub-user to modify the permission of the user. [Table 7-13](#) and [Table 7-14](#) list the detailed permission descriptions.

If the **Set Permission** button is gray, you do not have the permission to modify the package group or package. You can apply to the administrator, group owner, or other users who have the permissions on granting and revoking permissions of package groups or packages.

Revoking Permissions on Package Groups and Packages

Click **Revoke Permission** in the **Operation** column of a sub-user to revoke the user's permissions. After the operation, the sub-user does not have any permission on the package group or package.

Permissions Description

- Package group permissions
Querying permissions. A group owner can view the created package group and all packages in the group, and can also view package groups on which he or she has all permissions.
A package group is a unit. If you select a group when creating a package, you can grant only the permissions of the package group to other users.
- Package permissions
Querying permissions. A package owner can view the created packages, and can also view packages on which he or she has all permissions.

7.2.3 Creating a Package

DLI allows you to submit program packages in batches to the general-use queue for running.

NOTE

If you need to update a package, you can use the same package or file to upload it to the same location (in the same group) on DLI to overwrite the original package or file.

Procedure

1. On the left of the management console, choose **Data Management > Package Management**.
2. On the **Package Management** page, click **Create** in the upper right corner to create a package.
3. In the displayed **Create Package** dialog box, set related parameters by referring to [Table 7-15](#).

Table 7-15 Parameter description

Parameter	Description
Package Type	The following package types are supported: <ul style="list-style-type: none">• JAR: JAR file• PyFile: User Python file• File: User file• ModelFile: User AI model file
Package File Path	Select the OBS path of the corresponding packages. NOTE <ul style="list-style-type: none">• The packages must be uploaded to OBS for storage in advance.• Only files can be selected.
Group Policy	You can select Use existing group , Use new group , or No grouping .
Group Name	<ul style="list-style-type: none">• Use existing group: Select an existing group.• Use new group: Enter a custom group name.• No grouping: No need to select or enter a group name. NOTE <ul style="list-style-type: none">• If you select a group, the permission management refers to the permissions of the corresponding package group.• If no group is selected, the permission management refers to the permissions of the corresponding package. For details about how to manage permissions on package groups and packages, see Managing Permissions on Packages and Package Groups .

4. Click **OK**.

After a package is created, you can view and select the package for use on the **Package Management** page.

7.2.4 Deleting a Package

You can delete a package based on actual conditions.

Procedure

1. On the left of the management console, choose **Data Management > Package Management**.
2. Click **Delete** in the **Operation** column of the package to be deleted.
3. In the dialog box that is displayed, click **OK**.

7.2.5 Modifying the Owner

To change the owner of a package, click **Modify Owner** in the **Operation** column of a package on the **Package Management** page.

Table 7-16 Description

Parameter	Description
Group Name	<ul style="list-style-type: none">• If you select a group when creating a package, the name of the group is displayed.• If no group is selected when creating a package, this parameter is not displayed.
Name	Name of a package.
Select Type	<ul style="list-style-type: none">• If you select a group when creating a package, you can change the owner of the group or package.• If no group is selected when creating a package, this parameter is not displayed.
Username	Name of the package owner. NOTE The username is the name of an existing IAM user.

7.2.6 Built-in Dependency Packages

DLI built-in dependencies are provided by the platform by default. In case of conflicts, you do not need to upload them when packaging JAR packages of user-defined Spark or Flink jobs.

Spark 2.3.2 Dependency Packages

- accessors-smart-1.2.jar
- activation-1.1.1.jar
- aircompressor-0.8.jar
- alluxio-2.3.1-hw-luxor-SNAPSHOT-client.jar
- antlr-2.7.7.jar
- antlr4-runtime-4.8-1.jar
- antlr-runtime-3.4.jar
- aopalliance-1.0.jar
- aopalliance-repackaged-2.4.0-b34.jar
- apache-log4j-extras-1.2.17.jar
- arpack_combined_all-0.1.jar
- arrow-format-0.8.0.jar
- arrow-memory-0.8.0.jar
- arrow-vector-0.8.0.jar
- asm-5.0.4.jar
- audience-annotations-0.5.0.jar
- automaton-1.11-8.jar
- avro-1.7.7.jar

- avro-ipc-1.7.7.jar
- avro-ipc-1.7.7-tests.jar
- avro-mapred-1.7.7-hadoop2.jar
- aws-java-sdk-bundle-1.11.271.jar
- base64-2.3.8.jar
- bcpkix-jdk15on-1.66.jar
- bcprov-jdk15on-1.66.jar
- bonecp-0.8.0.RELEASE.jar
- breeze_2.11-0.13.2.jar
- breeze-macros_2.11-0.13.2.jar
- calcite-avatica-1.2.0-incubating.jar
- calcite-core-1.2.0-incubating.jar
- calcite-linq4j-1.2.0-incubating.jar
- carbondata-bloom-1.6.1.1032-2.0.0.dli-20201012.110925-6.jar
- carbondata-cli-1.6.1.1032-hw-2.0.0.dli-20201012.110947-6.jar
- carbondata-common-1.6.1.1032-hw-2.0.0.dli-20201012.110840-6.jar
- carbondata-common-plugin-1.6.1.1032-hw-2.0.0.dli-20201012.110945-6.jar
- carbondata-core-1.6.1.1032-hw-2.0.0.dli-20201012.110843-6.jar
- carbondata-format-1.6.1.1032-hw-2.0.0.dli-20201012.110953-12.jar
- carbondata-hadoop-1.6.1.1032-hw-2.0.0.dli-20201012.110849-6.jar
- carbondata-hive-1.6.1.1032-hw-2.0.0.dli-20201012.110911-6.jar
- carbondata-mv-core-1.6.1.1032-hw-2.0.0.dli-20201012.110936-6.jar
- carbondata-mv-plan-1.6.1.1032-hw-2.0.0.dli-20201012.110933-6.jar
- carbondata-processing-1.6.1.1032-hw-2.0.0.dli-20201012.110846-6.jar
- carbondata-secondary_index-1.6.1.1032-hw-2.0.0.dli-20201012.110942-6.jar
- carbondata-spark2-1.6.1.1032-hw-2.0.0.dli-20201012.110857-6.jar
- carbondata-spark-common-1.6.1.1032-hw-2.0.0.dli-20201012.110903-6.jar
- carbondata-spark-datasource-1.6.1.1032-hw-2.0.0.dli-20201012.110900-6.jar
- carbondata-store-sdk-1.6.1.1032-hw-2.0.0.dli-20201012.110852-6.jar
- carbondata-streaming-1.6.1.1032-hw-2.0.0.dli-20201012.110922-6.jar
- carbondata-vector-1.6.1.1032-hw-2.0.0.dli-20201012.110956-6.jar
- checker-qual-2.11.1.jar
- chill_2.11-0.8.4.jar
- chill-java-0.8.4.jar
- commons-beanutils-1.9.4.jar
- commons-cli-1.2.jar
- commons-codec-2.0-20130428.202122-59.jar
- commons-collections-3.2.2.jar
- commons-collections4-4.2.jar
- commons-compiler-3.0.8.jar

- commons-compress-1.4.1.jar
- commons-configuration2-2.1.1.jar
- commons-crypto-1.0.0-hw-20191105.jar
- commons-daemon-1.0.13.jar
- commons-dbcp-1.4.jar
- commons-dbcp2-2.7.0.jar
- commons-httpclient-3.1.jar
- commons-io-2.5.jar
- commons-lang-2.6.jar
- commons-lang3-3.5.jar
- commons-logging-1.2.jar
- commons-math3-3.4.1.jar
- commons-net-2.2.jar
- commons-pool-1.5.4.jar
- commons-pool2-2.8.0.jar
- commons-text-1.3.jar
- compress-lzf-1.0.3.jar
- core-1.1.2.jar
- curator-client-4.2.0.jar
- curator-framework-4.2.0.jar
- curator-recipes-2.7.1.jar
- datanucleus-api-jdo-3.2.6.jar
- datanucleus-core-3.2.10.jar
- datanucleus-rdbms-3.2.9.jar
- derby-10.12.1.1.jar
- dnsjava-2.1.7.jar
- ehcache-3.3.1.jar
- eigenbase-properties-1.1.5.jar
- error_prone_annotations-2.3.4.jar
- failureaccess-1.0.1.jar
- fastutil-8.2.3.jar
- ffmpeg-4.3.1-1.5.4.jar
- ffmpeg-4.3.1-1.5.4-linux-x86_64.jar
- flatbuffers-1.2.0-3f79e055.jar
- generex-1.0.2.jar
- geronimo-jcache_1.0_spec-1.0-alpha-1.jar
- gson-2.2.4.jar
- guava-29.0-jre.jar
- guice-4.0.jar
- guice-servlet-4.0.jar

- hadoop-annotations-3.1.1-hw-ei-302002.jar
- hadoop-auth-3.1.1-hw-ei-302002.jar
- hadoop-aws-3.1.1-hw-ei-302002.jar
- hadoop-client-3.1.1-hw-ei-302002.jar
- hadoop-common-3.1.1-hw-ei-302002.jar
- hadoop-hdfs-3.1.1-hw-ei-302002.jar
- hadoop-hdfs-client-3.1.1-hw-ei-302002.jar
- hadoop-mapreduce-client-common-3.1.1-hw-ei-302002.jar
- hadoop-mapreduce-client-core-3.1.1-hw-ei-302002.jar
- hadoop-mapreduce-client-jobclient-3.1.1-hw-ei-302002.jar
- hadoop-minikdc-3.1.1-hw-ei-302002.jar
- hadoop-yarn-api-3.1.1-hw-ei-302002.jar
- hadoop-yarn-client-3.1.1-hw-ei-302002.jar
- hadoop-yarn-common-3.1.1-hw-ei-302002.jar
- hadoop-yarn-registry-3.1.1-hw-ei-302002.jar
- hadoop-yarn-server-common-3.1.1-hw-ei-302002.jar
- hadoop-yarn-server-web-proxy-3.1.1-hw-ei-302002.jar
- hamcrest-core-1.3.jar
- HikariCP-java7-2.4.12.jar
- hive-common-1.2.1-hw-2.1.0.dli-20201111.064115-91.jar
- hive-exec-1.2.1-hw-2.1.0.dli-20201111.064444-91.jar
- hive-metastore-1.2.1-hw-2.1.0.dli-20201111.064230-91.jar
- hk2-api-2.4.0-b34.jar
- hk2-locator-2.4.0-b34.jar
- hk2-utils-2.4.0-b34.jar
- hppc-0.7.2.jar
- htrace-core4-4.2.0-incubating-hw-1.0.0.jar
- httpclient-4.5.4.jar
- httpcore-4.4.7.jar
- ivy-2.4.0.jar
- j2objc-annotations-1.3.jar
- jackson-annotations-2.10.0.jar
- jackson-core-2.10.0.jar
- jackson-core-asl-1.9.13-atlassian-4.jar
- jackson-databind-2.10.0.jar
- jackson-dataformat-yaml-2.10.0.jar
- jackson-datatype-jsr310-2.10.3.jar
- jackson-jaxrs-base-2.10.3.jar
- jackson-jaxrs-json-provider-2.10.3.jar
- jackson-mapper-asl-1.9.13-atlassian-4.jar

- jackson-module-jaxb-annotations-2.10.3.jar
- jackson-module-paranamer-2.10.0.jar
- jackson-module-scala_2.11-2.10.0.jar
- jakarta.activation-api-1.2.1.jar
- jakarta.xml.bind-api-2.3.2.jar
- janino-3.0.8.jar
- javacpp-1.5.4.jar
- javacpp-1.5.4-linux-x86_64.jar
- javacv-1.5.4.jar
- JavaEWAH-1.1.7.jar
- javassist-3.18.1-GA.jar
- javax.annotation-api-1.2.jar
- javax.inject-1.jar
- javax.inject-2.4.0-b34.jar
- javax.servlet-api-3.1.0.jar
- javax.ws.rs-api-2.0.1.jar
- java-xmlbuilder-1.1.jar
- javolution-5.3.1.jar
- jaxb-api-2.2.11.jar
- jcip-annotations-1.0-1.jar
- jcl-over-slf4j-1.7.26.jar
- jdo-api-3.0.1.jar
- jersey-client-2.23.1.jar
- jersey-common-2.23.1.jar
- jersey-container-servlet-2.23.1.jar
- jersey-container-servlet-core-2.23.1.jar
- jersey-guava-2.23.1.jar
- jersey-media-jaxb-2.23.1.jar
- jersey-server-2.23.1.jar
- jets3t-0.9.4.jar
- jetty-http-9.4.31.v20200723.jar
- jetty-io-9.4.31.v20200723.jar
- jetty-security-9.4.31.v20200723.jar
- jetty-server-9.4.31.v20200723.jar
- jetty-servlet-9.4.31.v20200723.jar
- jetty-util-9.4.31.v20200723.jar
- jetty-util-ajax-9.4.31.v20200723.jar
- jetty-webapp-9.4.31.v20200723.jar
- jetty-xml-9.4.31.v20200723.jar
- joda-time-2.9.3.jar

- jodd-core-4.2.0.jar
- json-20200518.jar
- json4s-ast_2.11-3.2.11.jar
- json4s-core_2.11-3.2.11.jar
- json4s-jackson_2.11-3.2.11.jar
- json-sanitizer-1.2.1.jar
- json-smart-2.3.jar
- jsp-api-2.1.jar
- jsr305-3.0.2.jar
- jta-1.1.jar
- jtransforms-2.4.0.jar
- jul-to-slf4j-1.7.26.jar
- junit-4.11.jar
- kerb-admin-1.0.1.jar
- kerb-client-1.0.1.jar
- kerb-common-1.0.1.jar
- kerb-core-1.0.1.jar
- kerb-crypto-1.0.1.jar
- kerb-identity-1.0.1.jar
- kerb-server-1.0.1.jar
- kerb-simplekdc-1.0.1.jar
- kerb-util-1.0.1.jar
- kerby-asn1-1.0.1.jar
- kerby-config-1.0.1.jar
- kerby-pkix-1.0.1.jar
- kerby-util-1.0.1.jar
- kerby-xdr-1.0.1.jar
- kryo-shaded-3.0.3.jar
- kubernetes-client-4.9.2-hw-20200804.jar
- kubernetes-model-4.9.2-hw-20200804.jar
- kubernetes-model-common-4.9.2-hw-20200804.jar
- leveledbjni-all-1.8-hw-20191105.jar
- libfb303-0.9.3.jar
- libthrift-0.12.0.jar
- listenablefuture-9999.0-empty-to-avoid-conflict-with-guava.jar
- log4j-1.2.17-cloudera1.jar
- log4j-rolling-appender-20131024-2017.jar
- logging-interceptor-3.14.4.jar
- luxor-encrypt-2.1.0-20201106.065437-53.jar
- luxor-fs3-2.1.0-20201106.065612-53.jar

- luxor-obs-fs3-2.1.0-20201106.065616-53.jar
- luxor-rpc_2.11-2.1.0-20201106.065541-53.jar
- luxor-rpc-protobuf2-2.1.0-20201106.065551-53.jar
- lz4-java-1.7.1.jar
- machinist_2.11-0.6.1.jar
- macro-compatible_2.11-1.1.1.jar
- metrics-core-3.1.5.jar
- metrics-graphite-3.1.5.jar
- metrics-jmx-4.1.12.1.jar
- metrics-json-3.1.5.jar
- metrics-jvm-3.1.5.jar
- minlog-1.3.0.jar
- mssql-jdbc-6.2.1.jre7.jar
- netty-3.10.6.Final.jar
- netty-all-4.1.51.Final.jar
- nimbus-jose-jwt-8.19.jar
- objenesis-2.1.jar
- okhttp-3.14.4.jar
- okio-1.17.2.jar
- opencsv-2.3.jar
- opencsv-4.6.jar
- opencv-4.3.0-2.jar
- orc-core-1.4.4-nohive.jar
- orc-mapreduce-1.4.4-nohive.jar
- oro-2.0.8.jar
- osgi-resource-locator-1.0.1.jar
- paranamer-2.8.jar
- parquet-column-1.8.3.jar
- parquet-common-1.8.3.jar
- parquet-encoding-1.8.3.jar
- parquet-format-2.3.1.jar
- parquet-hadoop-1.8.3.jar
- parquet-hadoop-bundle-1.6.0.jar
- parquet-jackson-1.8.3.jar
- parquet-format-2.3.1.jar
- parquet-hadoop-1.8.3.jar
- parquet-hadoop-bundle-1.6.0.jar
- parquet-jackson-1.8.3.jar
- postgresql-42.2.14.jar
- protobuf-java-2.5.0.jar

- py4j-0.10.7.jar
- pyrolite-4.13.jar
- re2j-1.1.jar
- RoaringBitmap-0.5.11.jar
- scala-compiler-2.11.12.jar
- scala-library-2.11.12.jar
- scalap-2.11.0.jar
- scala-parser-combinators_2.11-1.1.0.jar
- scala-reflect-2.11.12.jar
- scala-xml_2.11-1.0.5.jar
- secComponentApi-1.0.5c.jar
- shapeless_2.11-2.3.2.jar
- slf4j-api-1.7.30.jar
- slf4j-log4j12-1.7.30.jar
- snakeyaml-1.24.jar
- snappy-java-1.1.7.5.jar
- spark-catalyst_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.073826-143.jar
- spark-core_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.073836-134.jar
- spark-graphx_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.073847-129.jar
- spark-hive_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.073854-132.jar
- spark-kubernetes_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.073916-85.jar
- spark-kvstore_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.073933-127.jar
- spark-launcher_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.073940-127.jar
- spark-mllib_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.073946-127.jar
- spark-mllib-local_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.073953-127.jar
- spark-network-common_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.073959-127.jar
- spark-network-shuffle_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.074007-127.jar
- spark-om_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.074019-125.jar
- spark-repl_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.074028-125.jar
- spark-sketch_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.074035-125.jar
- spark-sql_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.074041-126.jar
- spark-streaming_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.074100-123.jar
- spark-tags_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.074136-123.jar
- spark-tags_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.074141-124-tests.jar
- spark-unsafe_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.074144-123.jar
- spark-uquery_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.074906-210.jar
- spark-yarn_2.11-2.3.2.0101-hw-2.1.0.dli-20201111.074151-123.jar
- spire_2.11-0.13.0.jar
- spire-macros_2.11-0.13.0.jar

- ST4-4.3.1.jar
- stax2-api-3.1.4.jar
- stax-api-1.0-2.jar
- stream-2.7.0.jar
- stringtemplate-3.2.1.jar
- token-provider-1.0.1.jar
- univocity-parsers-2.5.9.jar
- validation-api-1.1.0.Final.jar
- woodstox-core-5.0.3.jar
- xbean-asm5-shaded-4.4.jar
- xercesImpl-2.12.0.jar
- xml-apis-1.4.01.jar
- xz-1.0.jar
- zjsonpatch-0.3.0.jar
- zookeeper-3.5.6-hw-ei-302002.jar
- zookeeper-jute-3.5.6-hw-ei-302002.jar
- zstd-jni-1.4.4-11.jar

Flink 1.7.2 Dependency Packages

- bcpkix-jdk15on-1.60.jar
- bcprov-jdk15on-1.60.jar
- commons-codec-1.9.jar
- commons-configuration-1.7.jar
- deeplearning4j-core-0.9.1.jar
- deeplearning4j-nlp-0.9.1.jar
- deeplearning4j-nn-0.9.1.jar
- ejml-cdense-0.33.jar
- ejml-core-0.33.jar
- ejml-ddense-0.33.jar
- ejml-dsparse-0.33.jar
- ejml-experimental-0.33.jar
- ejml-fdense-0.33.jar
- ejml-simple-0.33.jar
- ejml-zdense-0.33.jar
- elsa-3.0.0-M7.jar
- esdk-obs-java-3.1.3.jar
- flink-cep_2.11-1.7.0.jar
- flink-cep-scala_2.11-1.7.0.jar
- flink-dist_2.11-1.7.0.jar
- flink-gelly_2.11-1.7.0.jar

- flink-gelly-scala_2.11-1.7.0.jar
- flink-ml_2.11-1.7.0.jar
- flink-python_2.11-1.7.0.jar
- flink-queryable-state-runtime_2.11-1.7.0.jar
- flink-shaded-curator-1.7.0.jar
- flink-shaded-hadoop2-uber-1.7.0.jar
- flink-table_2.11-1.7.0.jar
- guava-26.0-jre.jar
- hadoop-cloud-3.1.1-41-20201014.085840-4.jar
- httpasyncclient-4.1.2.jar
- httpclient-4.5.12.jar
- httpcore-4.4.4.jar
- httpcore-nio-4.4.4.jar
- java-xmlbuilder-1.1.jar
- jna-4.1.0.jar
- libtensorflow-1.12.0.jar
- log4j-api-2.8.2.jar
- log4j-core-2.8.2.jar
- log4j-over-slf4j-1.7.21.jar
- logback-classic-1.2.3.jar
- logback-core-1.2.3.jar
- nd4j-api-0.9.1.jar
- nd4j-native-0.9.1.jar
- nd4j-native-api-0.9.1.jar
- nd4j-native-platform-0.9.1.jar
- okhttp-3.14.8.jar
- okio-1.14.0.jar
- slf4j-api-1.7.21.jar
- tensorflow-1.12.0.jar

Python 3.7.5 Dependency Packages

- absl-py-0.8.1
- astor-0.8.0
- backports.functools_lru_cache-1.5
- backports.weakref-1.0.post1
- blaze-0.10.1
- Click-7.0
- contextlib2-0.6.0.post1
- cycler-0.10.0
- dataclasses-0.6

- datashape-0.5.2
- decorator-4.4.0
- Flask-1.1.1
- Flask_Cors-3.0.8
- Flask-1.16.0
- funcsigns-1.0.2
- gast-0.2.2
- google_pasta-0.1.7
- grpcio-1.24.1
- h5py-2.10.0
- itsdangerous-1.1.0
- Jinja2-2.10.3
- joblib-0.14.0
- Keras-2.3.1
- Keras_Applications-1.0.8
- Keras_Preprocessing-1.1.0
- kiwisolver-1.1.0
- Markdown-3.1.1
- MarkupSafe-1.1.1
- matplotlib-3.1.1
- mock-3.0.5
- multipledispatch-0.6.0
- multipledispatch2-0.4.0
- networkx-2.2
- numpy-1.16.1
- odo-0.5.0
- opt_einsum-3.1.0
- pandas-0.24.1
- pip-19.2.3
- protobuf-3.10.0
- psutil-5.6.3
- pyarrow-0.11.1
- pyparsing-2.4.2
- python-dateutil-2.8.0
- pytz-2019.3
- PyYAML-5.1.2
- scikit-learn-0.21.3
- scipy-1.1.0
- setuptools-41.2.0
- six-1.12.0

- python-dateutil-2.8.1
- pytz-2020.4
- PyYAML-3.12
- rpm-4.15.1
- scikit-learn-0.22.1
- scipy-1.5.4
- setuptools-47.1.1
- six-1.15.0
- tensorboard-1.13.1
- tensorflow-1.13.1
- tensorflow-estimator-1.13.0
- termcolor-1.1.0
- Werkzeug-1.0.1
- wheel-0.34.2
- zipp-3.4.0

8 Job Templates

8.1 SQL Template Management

To facilitate SQL operation execution, DLI allows you to customize query templates or save the SQL statements in use as templates. After templates are saved, you do not need to compile SQL statements. Instead, you can directly perform the SQL operations using the templates.

SQL templates include sample templates and custom templates. The default sample template contains 22 standard TPC-H query statements, which can meet most TPC-H test requirements.

SQL template management provides the following functions:

- [Sample Templates](#)
- [Custom Templates](#)
- [Creating a Template](#)
- [Executing the Template](#)
- [Searching for a Template](#)
- [Modifying a Template](#)
- [Deleting a Template](#)

Table Settings

In the upper right corner of the **SQL Template** page, click **Set Property** to determine whether to display templates by group.

If you select **Display by Group**, the following display modes are available:

- Expand the first group
- Expand all
- Collapse All

Sample Templates

The current sample template contains 22 standard TPC-H query statements. You can view the template name, description, and statements.

Table 8-1 Template management parameters

Parameter	Description
Name	Indicates the template name. <ul style="list-style-type: none">• A template name can contain only digits, letters, and underscores (_), but cannot start with an underscore (_) or contain only digits. It cannot be left empty.• The template name can contain a maximum of 50 characters.
Description	Description of the template you create.
Statement	SQL statement created as the template.
Operation	Execute: After you click this button, the system switches to the SQL Editor page, where you can modify or directly perform the statement as required. For details, see Executing the Template .

The existing sample templates apply to the following scenarios:

- Price summary report query
- Minimum cost supplier analysis
- Shipping priority analysis
- Analysis of order priority check
- Analysis of the number of local suppliers
- Analysis of forecasted income changes
- Freight volume analysis
- National market share analysis
- Profit estimation analysis by product type
- Analysis of returned parts
- Analysis of key inventory indicators
- Freight mode and command priority analysis
- Consumer allocation analysis
- Promotion effect analysis
- Analysis of the supplier with the largest contribution
- Analysis of the relationship between parts and suppliers

- Revenue analysis of small-lot orders
- Customer analysis for large orders
- Discounted revenue analysis
- Analysis of potential component improvements
- Analysis of suppliers who fail to deliver goods on time
- Global sales opportunity analysis

Custom Templates

The custom template list displays all templates you have created. You can view the template name, description, statements, and more.

Table 8-2 Template management parameters

Parameter	Description
Name	Indicates the template name. <ul style="list-style-type: none">• A template name can contain only digits, letters, and underscores (_), but cannot start with an underscore (_) or contain only digits. It cannot be left empty.• The template name can contain a maximum of 50 characters.
Description	Description of the template you create.
Statement	SQL statement created as the template.
Operation	<ul style="list-style-type: none">• Execute: After you click this button, the system switches to the SQL Editor page, where you can modify or directly perform the statement as required. For details, see Executing the Template.• Modify: Click Modify. In the displayed Modify Template dialog box, modify the template information as required. For details, see Modifying a Template.

Creating a Template

You can create a template on either the **Job Templates** or the **SQL Editor** page.

- To create a template on the **Job Templates** page:
 - a. On the left of the management console, choose **Job Templates > SQL Templates**.
 - b. On the **SQL Templates** page, click **Create Template** to create a template.
Enter the template name, SQL statement, and description information. For details, see [Table 8-3](#).

Table 8-3 Parameter description

Parameter	Description
Name	Indicates the template name. <ul style="list-style-type: none">• A template name can contain only digits, letters, and underscores (_), but cannot start with an underscore (_) or contain only digits. It cannot be left empty.• The template name can contain a maximum of 50 characters.
Statement	SQL statement to be saved as a template.
Description	Description of the template you create.
Group	<ul style="list-style-type: none">• Use existing• Use new• Do not use
Group Name	If you select Use existing or Use new , you need to enter the group name.

- c. Click **OK**.
- To create a template on the **SQL Editor** page:
 - a. On the left of the management console, click **SQL Editor**.
 - b. In the SQL job editing area of the displayed **SQL Editor** page, click **More** in the upper right corner and choose **Save as Template**.
Enter the template name, SQL statement, and description information.
For details, see [Table 8-3](#).
 - c. Click **OK**.

Executing the Template

You can execute a template on either the **Job Templates** or the **SQL Editor** page.

- To execute the template on the **Job Templates** page:
 - a. On the left of the management console, choose **Job Templates > SQL Templates**.
 - b. On the **SQL Templates** page, select a template and click **Execute** in the **Operation** column. The **SQL Editor** page is displayed, and the corresponding SQL statement is automatically entered in the SQL job editing window.
 - c. In the upper right corner of the SQL job editing window, Click **Execute** to run the SQL statement. After the execution is complete, you can view the execution result below the current SQL job editing window.
- To execute the template on the **SQL Editor** page:
 - a. On the left of the management console, click **SQL Editor**.
 - b. In the SQL job editing area of the displayed **SQL Editor** page, click **More** in the upper right corner and select **Choose Template**.

- c. In the displayed **Choose Template** dialog box, click the corresponding SQL statement template. The SQL job editing window is automatically displayed.
- d. Click **Execute** to run the SQL statement. After the execution is complete, you can view the execution result in the area below the current SQL job editing window.

Searching for a Template

On the **SQL Templates** page, you can enter the template name keyword in the search box on the upper right corner to search for the desired template.

Modifying a Template

- Step 1** On the **SQL Templates** page, locate the row where the target template resides and click **Modify** in the **Operation** column.
- Step 2** In the displayed **Modify Template** dialog box, modify the template name, statement, and description as required.
- Step 3** Click **OK**.

----End

Deleting a Template

On the **SQL Templates** page, select one or more templates to be deleted and click **Delete** to delete the selected templates.

8.2 Flink Template Management

Flink templates include sample templates and custom templates. You can modify an existing sample template to meet the actual job logic requirements and save the time for editing SQL statements. You can also customize a job template based on your habits and methods so that you can directly invoke or modify the template in later jobs.

Flink template management provides the following functions:

- [Sample Templates](#)
- [Custom Templates](#)
- [Creating a Template](#)
- [Creating a Job Based on a Template](#)
- [Modifying a Template](#)
- [Deleting a Template](#)

Sample Templates

The sample template list displays existing sample job templates. [Table 1](#) describes the parameters in the sample template list.

Table 8-4 Sample Templates Parameters

Parameter	Description
Name	Name of a template, which has 1 to 64 characters and only contains letters, digits, hyphens (-), and underlines (_).
Description	Description of a template. It contains 0 to 512 characters.
Operation	Create Job: Create a job directly by using the template. After a job is created, the system switches to the Edit page under Job Management .

The existing sample templates apply to the following scenarios:

- Real-time ETL of Nginx access logs to the database
- Fake plate vehicle detection
- Geo-fencing alarm
- Vehicle yawing alarm
- Vehicle speeding alarm
- Stream-based random forest anomaly detection
- CloudTable-DLI Flink-CloudTable
- DIS-DLI Flink-CSS (Elasticsearch)
- DIS-DLI Flink-CloudTable (OpenTSDB)
- DIS-DLI Flink-CloudTable
- DIS-DLI Flink-DCS
- DIS-DLI Flink-DDS (MongoDB)
- DIS-DLI Flink-DIS
- DIS-DLI Flink-OBS-DWS
- DIS-DLI Flink-SMN
- Kafka-DLI Flink-Kafka
- OBS-DLI Flink-RDS
- Stream-Join-Table (DCS)
- Stream-Join-Table (RDS)

Custom Templates

All custom templates are displayed in the custom template list. [Table 1](#) describes parameters involved in the custom template list.

Table 8-5 Parameters involved in the custom template list

Parameter	Description
Name	Name of a template, which has 1 to 64 characters and only contains letters, digits, hyphens (-), and underlines (_).

Parameter	Description
Description	Description of a template. It contains 0 to 512 characters.
Created	Time when a template is created.
Updated	Latest time when a template is modified.
Operation	<ul style="list-style-type: none"> ● Edit: Modify a template that has been created. ● Create Job: Create a job directly by using the template. After a job is created, the system switches to the Edit page under Job Management. ● Delete: Delete a created template.

Creating a Template

You can create a template using any of the following methods:

- Creating a template on the **Template Management** page
 - In the left navigation pane of the DLI management console, choose **Job Templates > Flink Templates**.
 - Click **Create Template** in the upper right corner of the page. The **Create Template** dialog box is displayed.
 - Specify **Name** and **Description**.

Table 8-6 Template parameters


Parameter	Description
Name	Name of a template, which has 1 to 64 characters and only contains letters, digits, hyphens (-), and underlines (_). NOTE The template name must be unique.
Description	Description of a template. It contains 0 to 512 characters.

- Click **OK** to enter the **Edit** page.

The following describes the functions of the **Editing** page, and the order follows the numeric values marked on the image.

Table 8-7 Description

Function	Description
Name	You can modify the template name.
Description	You can modify the template description.

Function	Description
Saving Mode	<ul style="list-style-type: none"> Save Here: Save the modification to the current template. Save as New: Save the modification as a new template.
SQL statement editing area	In the area, you can enter detailed SQL statements to implement business logic. For details about how to compile SQL statements, see the <i>Data Lake Insight SQL Syntax Reference</i> .
Save	Save the modifications.
Create Job	Use the current template to create a job.
Format	Format SQL statements. After SQL statements are formatted, you need to compile SQL statements again.
	Change the editing box style (black or white background).

- e. In the SQL statement editing area, enter SQL statements to implement service logic. For details about how to compile SQL statements, see the *Data Lake Insight SQL Syntax Reference*.
 - f. After the SQL statement is edited, click **Save** in the upper right corner to complete the template creation.
 - g. (Optional) If you do not need to modify the template, click **Create Job** in the upper right corner to create a job based on the current template. For details about how to create a job, see [Creating a Flink SQL Job](#) and [Creating a Custom Flink Job](#).
- Creating a template based on an existing job template
 - a. In the left navigation pane of the DLI management console, choose **Job Templates > Flink Templates**. Click the **Custom Templates** tab.
 - b. In the row where the desired template is located in the custom template list, click **Edit** under **Operation** to enter the **Edit** page.
 - c. After the modification is complete, set **Saving Mode** to **Save as New**.
 - d. Click **Save** in the upper right corner to save the template as a new one.
 - Creating a template using a created job
 - a. In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.
 - b. Click **Create Job** in the upper right corner. The **Create Job** page is displayed.
 - c. Specify parameters as required.
 - d. Click **OK** to enter the **Edit** page.
 - e. After the SQL statement is compiled, click **Set as Template**.
 - f. In the **Set as Template** dialog box that is displayed, specify **Name** and **Description** and click **OK**.

- Creating a template based on the existing job
 - a. In the left navigation pane of the DLI management console, choose **Job Management > Flink Jobs**. The **Flink Jobs** page is displayed.
 - b. In the job list, locate the row where the job that you want to set as a template resides, and click **Edit** in the **Operation** column.
 - c. After the SQL statement is compiled, click **Set as Template**.
 - d. In the **Set as Template** dialog box that is displayed, specify **Name** and **Description** and click **OK**.

Creating a Job Based on a Template

You can create jobs based on sample templates or custom templates.

1. In the left navigation pane of the DLI management console, choose **Job Templates > Flink Templates**.
2. In the sample template list, click **Create Job** in the **Operation** column of the target template. For details about how to create a job, see [Creating a Flink SQL Job](#) and [Creating a Custom Flink Job](#).

Modifying a Template

After creating a custom template, you can modify it as required. The sample template cannot be modified, but you can view the template details.

1. In the left navigation pane of the DLI management console, choose **Job Templates > Flink Templates**. Click the **Custom Templates** tab.
2. In the row where the template you want to modify is located in the custom template list, click **Edit** in the **Operation** column to enter the **Edit** page.
3. In the SQL statement editing area, modify the SQL statements as required.
4. Set **Saving Mode** to **Save Here**.
5. Click **Save** in the upper right corner to save the modification.

Deleting a Template

You can delete a custom template as required. The sample templates cannot be deleted. Deleted templates cannot be restored. Exercise caution when performing this operation.

1. In the left navigation pane of the DLI management console, choose **Job Templates > Flink Templates**. Click the **Custom Templates** tab.
2. In the custom template list, select the templates you want to delete and click **Delete** in the upper left of the custom template list.

Alternatively, you can delete a template by performing the following operations: In the custom template list, locate the row where the template you want to delete resides, and click **Delete** in the **Operation** column.
3. In the displayed dialog box, click **OK**.

9 Datasource Connections

9.1 Datasource Connection and Cross-Source Analysis

DLI supports the datasource capability of the native Spark and extends it. With DLI datasource connection, you can access other data storage services through SQL statements, Spark jobs, and Flink jobs and import, query, analyze, and process data in the services.

Datasource Connections

Before using DLI to perform cross source analysis, you need to set up a datasource connection to enable the network between data sources.

The enhanced datasource connection uses VPC peering at the bottom layer to directly connect the VPC network between the DLI queue and the destination datasource. Data is exchanged in point-to-point mode.

NOTE

- Datasource connections cannot be created for the default queue.
- **VPC Administrator** permissions are required to use the VPC, subnet, route, VPC peering connection, and port for DLI datasource connections. You can set this parameter in [Service Authorization](#).

Cross-Source Analysis

The enhanced datasource connection supports all cross-source services implemented by DLI and implements access to self-built data sources by using UDFs, Spark jobs, and Flink jobs.

Currently, DLI supports datasource connection to the following data sources: CloudTable HBase, CloudTable OpenTSDB, CSS, DCS Redis, DDS Mongo, DIS, DMS, DWS, MRS HBase, MRS Kafka, MRS OpenTSDB, OBS, RDS MySQL, RDS PostGre, and SMN.

 **NOTE**

- To access a datasource connection table, you need to use the queue for which a datasource connection has been created.
- The preview function is not supported for datasource tables.

Datasource Connection Analysis Process

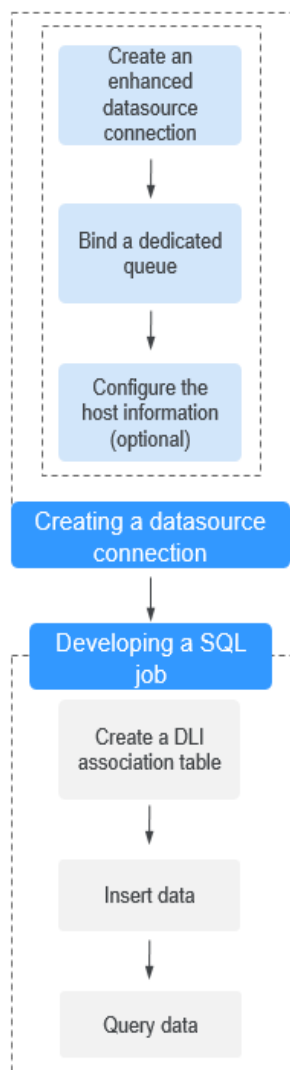
To use DLI for cross-source analysis, you need to create datasource connections and then develop different jobs to access data sources. Perform the following steps:

1. Create a datasource connection. You can create a connection in either of the following ways:
 - Create a datasource connection on the management console.
 - Create a datasource connection through an API.
2. Develop a DLI job to connect to a datasource. You can connect to a datasource in one of the following ways:
 - Develop a SQL job to connect to a datasource.
 - Develop a Spark job to connect to a datasource.
 - Develop a Flink job to connect to a datasource.

The following describes the basic processes of developing SQL jobs, Spark jobs, and Flink jobs for datasource connection.

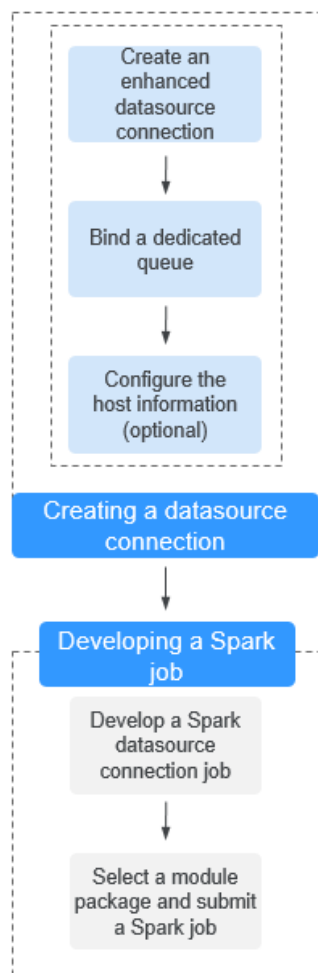
- SQL Job

Figure 9-1 Datasource connection analysis process

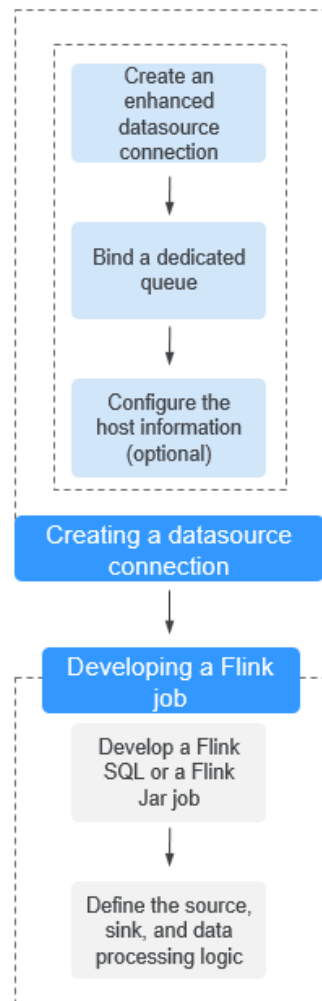


- Spark Job

Figure 9-2 Datasource connection analysis process



- Flink Job

Figure 9-3 Datasource connection analysis process

9.2 Enhanced Datasource Connections

The enhanced datasource connection uses VPC peering at the bottom layer to directly connect the VPC network between the DLI cluster and the destination datasource. Data is exchanged in point-to-point mode. The enhanced datasource connection function supports all cross-source services implemented by DLI, including HBase, CloudTable OpenTSDB, MRS OpenTSDB, DWS, RDS, CSS, DCS, and DDS. In addition, UDFs, Spark jobs, and Flink jobs can be used to access self-built data sources.

NOTE

- The CIDR block of the DLI queue which is bound with a datasource connection cannot overlap with that of the data source.
- Datasource connections cannot be created for the default queue.
- To access a datasource connection table, you need to use the queue for which a datasource connection has been created.
- The preview function is not supported for datasource tables.

It has the following functions:

- [Creating a Connection](#)
- [Searching for a Connection](#)
- [Viewing Connection Details](#)
- [Viewing Details about a Bound Queue](#)
- [Deleting a Connection](#)
- [Binding a Queue](#)
- [Unbinding a Queue](#)
- [Modifying the Host Information](#)
- [Custom Route Information](#)
- [Managing Permissions](#)

Enhanced Datasource Connection Page

This page displays all enhanced datasource connections. If there are a large number of connections, they are displayed on multiple pages.

Table 9-1 Datasource connection list parameters

Parameter	Description
Connection Name	Name of the created datasource connection.
Connection Status	Status of a datasource connection. Currently, the console displays only connections in the Active state.
VPC	The VPC used by the destination data source.
Subnet	The subnet used by the destination data source.
Created	Time when a connection is created. The connections in the connection list can be displayed according to the creation time in ascending or descending order.

Parameter	Description
Operation	<ul style="list-style-type: none">● Delete Connection: Delete a created datasource connection.● Routing Info: Used to display the custom route information of the datasource connection.● Modify Host: Customize the IP address corresponding to the host or domain name.● Bind Queue: After the binding queue is executed, the bound queue can use the enhanced datasource connection to the data source.● More:<ul style="list-style-type: none">– Unbind Queue: Unbind a datasource connection from a queue.– Add Route: Add a custom route for the datasource connection.– Delete Route: Delete a custom route for a datasource connection.– Manage Permissions: Authorize or reclaim permissions for other projects.

Creating a Connection

The following describes how to create a datasource HBase connection for MRS.

NOTE

Only enhanced datasource connection to MRS HBase is supported.

1. Apply for a cluster in MRS.
If a cluster is available, you do not need to apply for one.
2. In the navigation pane of the DLI management console, choose **Datasource Connections**.
3. Click the **Enhanced** tab and click **Create** in the upper left corner.
Enter the **Connection Name**, select the **Bind Queue** (optional), **VPC**, and **Subnet**, and enter the **Host Information** (optional). For details about the parameters, see [Table 9-2](#).

Table 9-2 Parameters

Parameter	Description
Connection Name	Name of the created datasource connection. <ul style="list-style-type: none">● The name can contain only letters, digits, and underscores (_). The parameter must be specified.● The connection name contains up to 64 characters.


Parameter	Description
Bind Queue	(Optional) Bind a queue that requires datasource connections. NOTE Before using an enhanced datasource connection, you must bind a queue and ensure that the VPC peering connection is in the Active state.
VPC	The VPC used by the destination data source.
Subnet	The subnet used by the destination data source.
Host Information	(Optional) The IP address corresponding to the host or domain name. Enter one record in each line. The format is as follows: IP address Host name/Domain name . When connecting to the HBase cluster of MRS, you need to configure the host name and IP address of the ZooKeeper instance. To go to the MRS cluster management page, perform the following steps: <ol style="list-style-type: none">1. In the MRS cluster list, click the name of the target cluster. The Overview page is displayed.2. In the Basic Information area of the cluster, locate the row that contains the target cluster and click View in the Operation column.3. Enter the username and password, and click Log In. (The username and password are set when you apply for an MRS cluster. The default username is admin.)4. Click the Service Management tab.5. Access the ZooKeeper service.6. Click the Instance tab to view the corresponding service IP address. You can select any service IP address. NOTE If the MRS cluster has multiple IP addresses, enter any service IP address when creating a datasource connection.

4. Click **OK**.

Searching for a Connection

On the **Enhanced** page, you can enter the keyword of a connection name in the search box to search for the matching connection.


Viewing Connection Details

On the **Enhanced** page, select a connection and click  to view its details. The connection ID and host information are included.

Viewing Details about a Bound Queue

On the **Enhanced** page, select a connection and click the connection name to view the information about the bound queue.

Table 9-3 Parameters in the datasource connection queue details list

Parameter	Description
VPC Peering ID	Specifies the ID of the VPC peering connection created in the cluster to which the queue belongs. NOTE A VPC peering connection is created for each queue bound to an enhanced datasource connection. The VPC peering connection is used for cross-VPC communication. Ensure that the security group used by the data source allows access from the DLI queue CIDR block, and do not delete the VPC peering connection during the datasource connection.
Queue	Name of a bound queue.
Connection Status	Datasource connection status. The following three statuses are available: <ul style="list-style-type: none">• Creating• Active• Failed NOTE If the connection status is Failed , click  on the left to view the detailed error information.
Updated	Time when a connection is updated. The connections in the connection list can be displayed according to the update time in ascending or descending order.
Operation	Unbind Queue: This operation is used to unbind a datasource connection from a queue.

Deleting a Connection

On the **Enhanced** page, click **Delete Connection** in the **Operation** column to delete unnecessary connections.

NOTE

A connection with **Connection Status** of **Creating** cannot be deleted.

Binding a Queue

- Method 1:
On the **Enhanced** page, select a connection and click **Bind Queue** in the **Operation** column. In the dialog box that is displayed, select the queues to be bound and click **OK**.

- Method 2:
Click the name of the selected connection. The **Connection Details** page is displayed. Click **Create** in the upper left corner. In the dialog box that is displayed, select the queues to be bound and click **OK**.

Unbinding a Queue

- Method 1:
On the **Enhanced** page, select a connection and click **More > Unbind Queue** in the **Operation** column. In the dialog box that is displayed, select the queues to be unbound and click **OK**.
- Method 2:
Click the name of the selected connection. The **Connection Details** page is displayed. Select the queue to be unbound and click **Unbind Queue** in the **Operation** column. In the displayed dialog box, confirm the queue to be unbound and click **OK**.

Modifying the Host Information

On the **Enhanced** page, select a connection and click **Modify Host**. In the dialog box that is displayed, enter the host information. The format is **Host IP address Host name/Domain name**. Multiple records are separated by line breaks.

NOTE

- After this configuration, all host information is overwritten.
- The host name or domain name can contain a maximum of 128 characters, including digits, letters, underscores (_), hyphens (-), and periods (.). It must start with a letter.

Custom Route Information

You can add custom routes to enhanced datasource connections.

- Viewing route information
On the **Enhanced** page, select a connection and choose **Routing Info** in the **Operation** column to view the route information of the datasource connection.
- Adding a route
On the **Enhanced** page, select a connection and choose **More > Add Route** in the **Operation** column, or click **Add Route** on the **Routing Info** page to add a custom route for the datasource connection. In the displayed dialog box, enter the route name and route CIDR block. For details about the parameters, see [Table 9-4](#).

Table 9-4 Parameters in the custom route details list

Parameter	Description
Connection Name	Name of the connection for which a route is to be added.

Parameter	Description
Route Name	Name of a custom route, which is unique in the same enhanced datasource scenario. The name contains 1 to 64 characters, including digits, letters, underscores (_), and hyphens (-).
IP Address	Custom route CIDR block. The CIDR block of different routes can overlap but cannot be the same.

- Deleting a route
On the **Enhanced** page, select a connection and choose **More > Delete Route** in the **Operation** column, or click **Delete Route** on the **Routing Info** page to delete a custom route for the datasource connection.

Managing Permissions

You can grant or revoke permissions for enhanced datasource connections of other projects.

- Grant permissions: In the **Manage Permissions** dialog box, select **Add IAM User** and enter the ID of the project to be authorized.

After a project is authorized, you can log in to the system as a user of the authorized project or switch to the corresponding project. On the **Enhanced** tab page, you can view the authorized datasource connection and bind the created queue to the datasource connection. Cross-project datasource connections and routes can be created.

NOTE

- If the authorized projects belong to different users in the same region, you can use the user account of the authorized projects to log in.
- If the authorized projects belong to the same user in the same region, you can use the current account to switch to the corresponding project.

For example, if project B needs to access the data source of project A, perform the following operations:

- For Project A:
 - i. Log in to DLI using the account of project A.
 - ii. Create an enhanced datasource connection **ds** in DLI based on the VPC information of the corresponding data source.
 - iii. Grant project B the permission to access the enhanced datasource connection **ds**.
- For Project B:
 - i. Log in to DLI using the account of project B.
 - ii. Bind the enhanced datasource connection **ds** to a queue.
 - iii. (Optional) Set host information and create a route.

After creating a VPC peering connection and route between the enhanced datasource connection of project A and the queue of project B, you can create a job in the queue of project B to access the data source of project A.

- Revoke: In the **Manage Permissions** dialog box, select **Revoke Permission** and select the ID of the project whose permissions need to be revoked.

9.3 Managing Datasource Connection Permissions

Scenarios

- You can isolate datasource connections allocated to different users by setting permissions to ensure data query performance.
- The administrator and datasource connection owner have all permissions, which cannot be set or modified by other users.

On the **Datasource Authentication** tab page, click **Manage Permissions** in the **Operation** column of the row that contains the authentication to be modified. On the **User Permission Info** page that is displayed, you can grant, set, and revoke permissions of the datasource connection.

Granting Permissions on Datasource Connections

Click **Grant Permission** in the upper right corner of the page.

Table 9-5 Permission granting parameters

Parameter	Description
Username	Name of the authorized IAM user. NOTE The username is the name of an existing IAM user.
Select the permissions to be granted to the user	<ul style="list-style-type: none">• Access: This permission allows you to access the datasource connection.• Update: This permission allows you to update the datasource connection.• Delete: This permission allows you to delete the datasource connection.• Grant Permission: This permission allows you to grant the datasource connection permission to other users.• Grant Permission: This permission allows you to revoke the datasource connection permission to other users. However, you cannot revoke the permissions of the datasource connection owner.• View Other User's Permissions: This permission allows you to view the datasource connection permissions of other users.

Setting Permissions

Click **Set Permission** in the **Operation** column of the sub-user to modify the permission of the user. [Table 9-5](#) lists the detailed permission descriptions.

If all options under **Set Permission** are gray, you are not allowed to change permissions on this datasource connection. You can apply for the granting and

revoking permissions from administrators, group owners, and other users who have the permission to grant permissions.

Revoking Permissions

Click **Revoke Permission** in the **Operation** column of a sub-user to revoke the user's permissions. After this operation, the sub-user does not have any permission of the datasource connection.

9.4 Datasource Authentication

Scenarios

Datasource authentication is used to manage authentication information for accessing CSS and MRS security environments and encrypt passwords for accessing DWS, RDS, DDS, and DCS data sources.

- When creating a CSS cluster in the security mode, you need to specify the username and password. A certificate is available for downloading the security CSS cluster. To access the security CSS cluster, the username, password, and certificate need to be stored in DLI. The username, password, and certificate are the datasource authentication information for accessing the CSS cluster.
- When creating an MRS cluster in the security mode, you need to enable the kerberos authentication. Download the authentication credential, which contains the **krb5.conf** and **user.keytab** files. When connecting DLI to an MRS cluster, you can store the two files in DLI. The two files are the datasource authentication information for accessing the MRS cluster.
- To create a Kafka cluster, you need to enable SSL access. You can configure this function on the **Service Configuration** page of MRS and Kafka, or set up an open-source Kafka cluster and modify the configuration file to enable this function.
- Through datasource password authentication, the passwords for accessing DWS, RDS, DDS, and DCS data sources can be encrypted for storage.

Creating a Datasource Authentication

1. Create a data source to be accessed.

NOTE

- If a cluster is available, you do not need to apply for one.
- Create a CSS cluster in the security mode: Select 6.5.4 or a later version for **Cluster Version** and enable the **Security Mode**.
 - Create an MRS cluster in the security mode: On the MRS management console, apply for a cluster, select **Custom Config**, set **Cluster Version** to MRS 2.1.0 or later, and enable **Kerberos Authentication**.
 - Creating a Kafka cluster: You can configure the Kafka cluster on the Service Configuration page of MRS and Kafka. For details about how to configure SSL on MRS, see **Security Configuration** in the *MapReduce Service User Guide*.

- Creating a DWS cluster: Apply for a data warehouse cluster from DWS. For details, see the *Data Warehouse Service Management Guide*.
 - Creating an RDS data source: Apply for a database instance in the RDS service. For details, see the *Relational Database Service Getting Started*.
 - Creating a DCS data source: Apply for a DCS instance in DCS. For details, see the *Distributed Cache Service User Guide*.
 - Creating a DDS data source: Apply for a DDS DB instance. For details, see *Document Database Service Getting Started*.
2. Download the authentication credential. Skip this step when you access DWS, RDS, DCS, or DDS data sources.
 - CSS security cluster: On the **Cluster Management** page, click the cluster name. On the **Basic Information** page that is displayed, find the **Security Mode** and download the certificate of the CSS security cluster.
 - MRS security cluster: On the cluster list page, click the cluster name. On the displayed cluster information page, download the authentication credential. After downloading the authentication credential, decompress it to obtain the **krb5.conf** and **user.keytab** files.
 3. Upload the authentication credential. Skip this step when you access DWS, RDS, DCS, or DDS data sources.

Upload the obtained authentication credential file to the user-defined OBS bucket.
 4. On the DLI management console, click **Datasource Connection**.
 5. On the **Datasource Authentication** tab page, click **Create** to create an authentication information.
 - CSS

Table 9-6 Parameters

Parameter	Description
Authentication Certificate	Name of the datasource authentication information to be created. <ul style="list-style-type: none">• The name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_).• The length of the database name cannot exceed 128 characters.• It is recommended that the name contain the CSS security cluster name to distinguish security authentication information of different clusters.
Type	Select CSS.
Username	Username for logging in to the security cluster.
Password	The password of the security cluster
Certificate Path	The OBS path for uploading the security certificate

- MRS

Table 9-7 Description

Parameter	Description
Authentication Certificate	Name of the datasource authentication information to be created. <ul style="list-style-type: none"> The name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_). The length of the database name cannot exceed 128 characters. It is recommended that the name contain the MRS security cluster name to distinguish security authentication information of different clusters.
Type	Select Kerberos .
Username	Username for logging in to the security cluster.
krb5_conf Path	OBS path to which the krb5.conf file is uploaded NOTE The renew_lifetime configuration item under [libdefaults] must be removed from krb5.conf . Otherwise, the Message stream modified (41) problem may occur.
keytab Path	OBS path to which the user.keytab file is uploaded

- Kafka

Table 9-8 Parameters

Parameter	Description
Authentication Certificate	Name of the datasource authentication information to be created. <ul style="list-style-type: none"> The name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_). The length of the database name cannot exceed 128 characters.
Type	Select Kafka SSL .
Truststore Path	OBS path to which the SSL truststore file is uploaded.
Truststore Password	Truststore password.
Keystore Path	OBS path to which the SSL keystore file (key and certificate) is uploaded.

Parameter	Description
Keystore Password	Keystore (key and certificate) password.
Key Password	Password of the private key in the keystore file.

- Password (datasource password authentication)
Create datasource authentication for accessing DWS, RDS, DCS, and DDS data sources.

Table 9-9 Parameters

Parameter	Description
Authentication Certificate	Name of the datasource authentication information to be created. <ul style="list-style-type: none">• The name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_).• The length of the database name cannot exceed 128 characters.
Type	Select Password .
Username	Username for accessing the datasource
Password	Password for accessing the datasource

Searching for a Datasource Authentication

On the **Datasource Authentication** tab, you can enter the authentication information name in the search box to search for the matching authentication information. To ensure user information security, the password field is not returned.

Updating Authentication Information

On the **Datasource Authentication** tab, click **Update** in the **Operation** column of the authentication information to be modified. Currently, only the username and password can be updated. If you need to update the certificate, delete the authentication information and create a new one.

NOTE

The username and password are optional. If they are not set, the field is not modified.

Deleting a Datasource Authentication

On the **Datasource Authentication** tab, click **Delete** in the **Operation** column of the authentication information to be deleted.

10 Global Configuration

10.1 Global Variables

Global variables can be used to simplify complex parameters. For example, long and difficult variables can be replaced to improve the readability of SQL statements.

Creating Variables

1. In the navigation pane of the DLI console, click **Global Variables**.
2. On the **Global Variables** page, click **Create** in the upper right corner to create a global variable.

Table 10-1 Parameters description

Parameter	Description
Variable	Name of the created global variable.
Value	Global variable value.

3. After creating a global variable, use **{{xxxx}}** in the SQL syntax to replace the parameter value set as the global variable. **xxxx** indicates the variable name. For example, in the table creation statement of DWS datasource connection, set the value **dbadmin.customer** of the **dbtable** parameter to the global variable **abc**. You can use **{{abc}}** to replace the actual variable value.

```
CREATE TABLE 'dws_test_var'  
USING dws  
OPTIONS (  
  'url' '*****', //Set the URL based on the site requirements.  
  'dbtable' '{{abc}}',  
  'user' 'dbadmin',  
  'password' '123'  
)
```

Modifying Variables

On the **Global Variables** page, click **Modify** in the **Operation** column of a variable to modify the variable value.

Deleting Variables

On the **Global Variables** page, click **Delete** in the **Operation** column of a variable to delete the variable value.

NOTE

- Only the user who creates a global variable can delete the variable.
- After a variable is deleted, the variable cannot be used in SQL statements.

10.2 Service Authorization

After entering the DLI management console, you are advised to set agency permissions to ensure that DLI can be used properly.

If you need to adjust the agency permissions, modify them on the **Service Authorization** page.

Currently, the following agency permissions need to be set for DLI:

- **Tenant Administrator (global): Tenant Administrator** permissions are required to access data from OBS to execute Flink jobs on DLI, for example, obtaining OBS/DWS data sources, log dump (including bucket authorization), checkpointing enabling, and job import and export.

NOTE

Due to cloud service cache differences, operations require about 60 minutes to take effect.

- **DIS Administrator** permissions are required to use DIS data as the data source of DLI Flink jobs.

NOTE

Due to cloud service cache differences, operations require about 30 minutes to take effect.

- To use CloudTable data as the data source of DLI Flink jobs, **CloudTable Administrator** permissions are required.

NOTE

Due to cloud service cache differences, operations require about 3 minutes to take effect.

- **VPC Administrator** permissions are required to use the VPC, subnet, route, VPC peering connection, and port for DLI datasource connections.

NOTE

Due to cloud service cache differences, operations require about 3 minutes to take effect.

- To receive notifications when a DLI job fails, **SMN Administrator** permissions are required.

 **NOTE**

Due to cloud service cache differences, operations require about 3 minutes to take effect.

- To use IEF to execute Flink Edge jobs in DLI, **Tenant Administrator** permissions are required. This permission is also required for running other services that require the **Tenant Administrator** permission.

 **NOTE**

Due to cloud service cache differences, operations require about 3 minutes to take effect.

After you agree to the authorization, DLI will create an agency named **dli_admin_agency** on the page.

 **NOTE**

- Only the tenant account or a subaccount of user group **admin** can authorize access.
- Do not delete **dli_admin_agency** after it is created successfully.

On this page, select or deselect the preceding permissions as required and click **Update**. If the system displays a message indicating that the agency permissions are updated successfully, the modification is successful.

11 Using UDFs

DLI allows you to query data by using the user-defined functions (Hive UDF).

Procedure

1. Compile a UDF.
Create or modify the content of **SumUdfDemo.java** in the sample code based on service requirements.
2. Generate a JAR package, set the output JAR package to **TestUDF.jar**, and run the **Build Artifacts** command.
A file named **TestUDF.jar** is generated in the corresponding path, for example, **udfDemo\target\artifacts\TestUDF**.
3. Upload **TestUDF.jar** to OBS. For details about how to upload data to OBS, see *Step 2: Upload Data to OBS* in [Submitting a SQL Job](#).
4. Create a function.
Run the following command on the management console to create a function:

```
CREATE FUNCTION fun1 AS 'com.demo.SumUdfDemo' using jar 'obs://udf/TestUDF.jar';
```
5. Use the created function.
Run the following statement to query using the function created in 4.

```
select fun1(ip) from ip_tables;
```
6. Delete the created function.
If this function is no longer used, run the following statement to delete the function:

```
Drop FUNCTION fun1;
```

Sample Code

The sample code in **SumUdfDemo.java** is as follows:

```
package com.demo;
import org.apache.hadoop.hive ql.exec.UDF;
public class SumUdfDemo extends UDF {
    public Int evaluate(Int a, Int b) {
        return a + b;
    }
}
```