Data Lake Insight

API Reference

Issue 01

Date 2024-08-19

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


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

Notice

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

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

Security Declaration

Vulnerability

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

Contents

1 Before You Start

1.1 Overview

Welcome to Data Lake Insight API Reference.

Data Lake Insight (DLI) is a serverless data processing and analysis service fully compatible with **Apache Spark**, **Trino**, and **Apache Flink** ecosystems. It frees you from managing any server. You can use stream analysis (with standard SQL), batch analysis (with Spark SQL), and interactive analysis (with Flink SQL) to query mainstream data formats without data ETL. DLI supports SQL statements and Spark applications for heterogeneous data sources, including RDS, GaussDB(DWS), CSS, OBS, custom databases on ECSs, and offline databases.

You can use APIs provided in this document to perform DLI operations, including queues, SQL jobs, Flink jobs, Spark jobs, and datasource connections. For details about all supported operations, see **Overview**.

Before calling DLI APIs, get yourself familiar with DLI concepts. For details, see **What Is DLI?**

1.2 API Calling

DLI provides RESTful (Representational State Transfer) APIs, allowing you to use HTTPS to call them. For details, see **Making an API Request**.

Unless otherwise specified, DLI APIs are synchronous. Specifically, **is_success**, indicating whether a request is successfully executed, is returned.

If the DLI API is asynchronous, run the API related to **Querying Job Status** according to the value of **job_id** in the response message to learn whether the API is successfully executed.

1.3 Endpoints

An endpoint is the **request address** for calling an API. Endpoints vary depending on services and regions. You need to select an endpoint based on your service requirements.

1.4 Constraints

- The number of resources that you can create is determined by your quota. To view or increase the quota, see **How Do I Increase the Image Quota?**
- For more constraints, see the API descriptions.

1.5 Basic Concepts

Account

An account is created when a user registers a Huawei ID and enables Huawei Cloud services. The account has full access permissions for all of its cloud services and resources. It can be used to reset user passwords and grant user permissions. The account is a payment entity and should not be used to perform routine management. For security purposes, create IAM users and grant them permissions for routine management.

User

An IAM user is created by an account in IAM to use cloud services. Each IAM user has its own identity credentials (password and access keys).

The account name, username, and password will be required for API authentication.

Region

Regions are divided based on geographical location and network latency. Public services, such as Elastic Cloud Server (ECS), Elastic Volume Service (EVS), Object Storage Service (OBS), Virtual Private Cloud (VPC), Elastic IP (EIP), and Image Management Service (IMS), are shared within the same region. Regions are classified as common regions and dedicated regions. A common region provides common cloud services that should be made available to all tenants. A dedicated region provides services of a specific type or only for specific tenants.

For details, see **Region and AZ**.

AZ

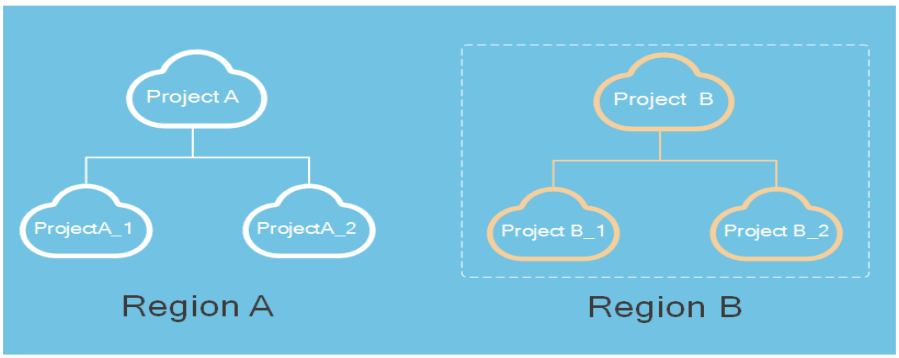
An AZ contains one or more physical data centers. Each AZ has independent cooling, fire extinguishing, moisture-proof, and electricity facilities. Within an AZ, computing, network, storage, and other resources are logically divided into multiple clusters. AZs within a region are interconnected using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

Project

Projects group and isolate compute, storage, and network resources across physical regions. A default project is provided for each region, and subprojects can be created under each default project. Users can be granted permissions to access

all resources in a specific project. If you need more refined access control, create subprojects under a default project and purchase resources in subprojects. Then you can assign users the permissions required to access only the resources in the specific subprojects.

Figure 1-1 Project isolating model



Enterprise Project

Enterprise projects group and manage resources across regions. Resources in enterprise projects are logically isolated from each other. An enterprise project can contain resources of multiple regions, and resources can be added to or removed from enterprise projects.

For details about how to obtain enterprise project IDs and features, see **Enterprise Management Service User Guide**.

2 Overview

This section describes the APIs provided by DLI.

Table 2-1 DLI APIs

Type	Description
Permission-related APIs	You can assign permissions to queues, view queue users, assign data permissions, view database users, view table users, view table user permissions, and view the permissions of authorized users.
Global variable-related APIs	You can create, delete, modify, and query global variables.
APIs related to enhanced datasource connections	You can create, delete, and query enhanced datasource connections, as well as bind and unbind queues, modify host information, and query enhanced datasource connection permissions.
Queue-related APIs (recommended)	You can create, delete, and query all queues, modify the CIDR block of a queue, restart, scale up, and scale down queues, view detailed information about queues, create and query connectivity test requests for specific addresses, create, query, modify, and delete scheduled scaling plans for queues.
SQL job-related APIs	You can import and export data, submit SQL jobs, cancel jobs, query all jobs, preview job results, query job status, query job details, check SQL syntax, and export query results.
SQL template-related APIs	You can perform operations on templates, such as saving SQL templates and creating job templates.

Type	Description
Flink job-related APIs	You can authorize DLI to OBS, create and update SQL jobs and user-defined Flink jobs, run jobs in batches, query the job list, job details, job execution plans, and job monitoring information. You can also stop jobs in batches, delete and batch delete jobs, export and import Flink jobs, create static flow diagrams for Flink SQL jobs, create IEF message channels, report Flink job status and callback Flink job actions at the edge, and report IEF system events.
APIs related to Flink job templates	You can create, update, and delete a template, and query the template list.
APIs related to Spark jobs	Creating batch jobs, cancel batch jobs, querying batch job lists, querying batch job details, querying batch job status, and querying batch job logs.
APIs related to Spark job templates	You can perform operations on templates, such as saving SQL templates and creating job templates.

3 Calling APIs

3.1 Making an API Request

This section describes the structure of a REST API request, and uses the IAM service as an example to explain **how to obtain a user token** to call an API. The obtained token can then be used to authenticate the calling of other APIs.

This section describes the structure of a REST API request, and uses the IAM service as an example to explain how to obtain a user token to call an API. The obtained token can then be used to authenticate the calling of other APIs.

Request URI

A request URI is in the following format:

{URI-scheme} :// {Endpoint} / {resource-path} ? {query-string}

Although a request URI is included in a request header, most programming languages or frameworks require the request URI to be separately transmitted, rather than being conveyed in a request message.

Table 3-1 Parameters in a URI

Parameter	Description
URI-scheme	Protocol used to transmit requests. All APIs use HTTPS.
Endpoint	Domain name or IP address of the server bearing the REST service.
resource-path	Access path of an API for performing a specified operation. Obtain the value from the URI of an API. For example, the **resource-path** of the API used to **obtain a user token** is **/v3/auth/tokens**.

Parameter	Description
query-string	Query parameter, which is optional. Ensure that a question mark (?) is included before each query parameter that is in the format of *Parameter name*=*Parameter value*. For example, **? limit=10** indicates that a maximum of 10 data records will be displayed.

NOTE

To simplify the URI display, each API is provided with only a **resource-path** and a request method. This is because the **URI-scheme** value of all APIs is **HTTPS**, and the endpoints in a region are the same. Therefore, the two parts are omitted.

Request Methods

HTTP-based request methods, which are also called operations or actions, specify the type of operations that you are requesting.

Table 3-2 HTTP method

Request Method	Description
GET	Requests a server to return specified resources.
PUT	Requests the server to update specified resources.
POST	Requests the server to add resources or perform special operations.
DELETE	Requests the server to delete a specific resource, for example, an object.
HEAD	Same as GET except that the server must return only the response header.
PATCH	Requests a server to update part of specified resources. If the resource is unavailable, the PATCH method is used to create a resource.

For example, in the case of the API used to **obtain a user token**, the request method is **POST**. The request is as follows:

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
```

Request Header

You can also add additional fields to a request, such as the fields required by a specified URI or an HTTP method. For example, add **Content-Type** that defines a request body type to request for the authentication information.

Table 3-3 lists common request header fields.

Table 3-3 Common request headers

Parameter	Description	Mandatory	Example
Host	Specifies the information about the requested server. The value can be obtained from the URL of the service API. The value is **hostname[:port]**. If the port number is not specified, the default port is used. The default port number for **https** is port **443**.	No This header field is mandatory if AK/SK authentication is in use.	code.test.com or code.test.com:443
Content-Type	Specifies the request body type or format. This field is mandatory and its default value is application/json. For other values, the description will be provided for specific APIs.	Yes	application/json
Content-Length	Indicates the length of the request body. The unit is byte.	This field is mandatory for POST and PUT requests, but must be left blank for GET requests.	3495
X-Project-ID	Project ID. It is mandatory in multi-project scenarios to obtain tokens for different projects.	No	e9993fc787d94b6c886cbaa340f9c0f4
X-Auth-Token	User token. User token is a response to the API for obtaining a user token (only this API does not require authentication).	This parameter is mandatory only for authentication using tokens.	The following is part of an example token: MIIPAgYJKoZIhvc-NAQcCo...ggg1BBIINPXsidG9rZ
Authorization	Specifies the signature authentication information. The value can be obtained from the request signing result.	This header field is mandatory if AK/SK authentication is in use.	-

Parameter	Description	Mandatory	Example
X-Sdk-Date	Specifies the time when the request is sent. The time is in *YYYYMMDD*'**T**'*HHMMSS*'**Z**' format. The value is the current Greenwich Mean Time (GMT) of the system.	This header field is mandatory if AK/SK authentication is in use.	20150907T101459Z
X-Language	Request language.	No	en-us

NOTE

In addition to supporting token-based authentication, APIs also support authentication using access key ID/secret access key (AK/SK). During AK/SK-based authentication, an SDK is used to sign the request, and the **Authorization** (signature authentication) and **X-Sdk-Date** (time when the request is sent) header fields are automatically added to the request.

The API for **obtaining a user token** does not require authentication. Therefore, this API only requires adding the Content-Type field. The request with the added Content-Type header is as follows:

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json
```

Request Body

A request body is generally sent in a structured format. It corresponds to **Content-Type** in the request header and transfers data except for the request header.

The request body varies according to the APIs. Certain APIs do not require the request body, such as the GET and DELETE APIs.

For the API of **obtaining a user token**, obtain the request parameters and parameter description from the API request. The following provides an example request with a body included. Replace ***username***, ***domainname***, ************* (login password), and ***xxxxxxxxxxxxxxxxx*** (project ID) with the actual values. Obtain a project ID from .

NOTE

scope specifies where a token takes effect. In the following example, the token takes effect only on the resources specified by the project ID. In the following example, the token takes effect only for the resources in a specified project. For more information about this API, see **Obtaining a User Token**.

```
POST https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/tokens
Content-Type: application/json

{
    "auth": {
        "identity": {
            "methods": [
                "password"
```



```
        ],
        "password": {
            "user": {
                "name": "username",
                "password": "********",
                "domain": {
                    "name": "domainname"
                }
            }
        }
    },
    "scope": {
        "project": {
            "name": "xxxxxxxxxxxxxxxxxx"
        }
    }
  }
}
```

If all data required by a request is available, you can send the request to call an API through **curl**, **Postman**, or coding. In the response to the IAM API used to obtain a user token, **x-subject-token** is the desired user token. Then, you can use the token to authenticate the calling of other APIs.

3.2 Authentication

API calling is authenticated using either of the following methods:

- Token authentication: Common requests are authenticated using Tokens.
- AK/SK authentication: Requests are encrypted using the access key ID and secret access key (AK/SK pair) to provide higher security. AK/SK authentication is recommended because it provides higher security than token authentication.

Token Authentication

A token specifies certain permissions in a computer system. Authentication using a token adds the token in a request as its header during API calling to obtain permissions to operate APIs through IAM.

NOTE

The validity period of a token is 24 hours. When using a token for authentication, cache it to prevent frequently calling the IAM API.

When calling the API to **obtain a user token**, you must set **auth.scope** in the request body to **project**.

```
{
  "auth": {
    "identity": {
      "methods": [
        "password"
      ],
      "password": {
        "user": {
          "name": "username",
          "password": "********",
          "domain": {
            "name": "domainname"
          }
        }
```



```
        }
      },
      "scope": {
         "project": {
            "name": "xxxxxxxx"
         }
      }
   }
}
```

After obtaining the token, add the **X-Auth-Token** header in a request to specify the token when calling other APIs. For example, if the token is **ABCDEFJ....**, add **X-Auth-Token: ABCDEFJ....** in a request as follows:

```
GET https://iam.ap-southeast-1.myhuaweicloud.com/v3/auth/projects
Content-Type: application/json
X-Auth-Token: ABCDEFJ....
```

AK/SK Authentication

In authentication using AK/SK, AK/SK is used to sign a request and add the signature in a request as its header for authentication.

NOTE

Authentication using AK/SK supports API requests with a body not larger than 12 MB. For API requests with a larger body, authentication using tokens is recommended.

- AK: access key ID, which is a unique identifier associated with a secret access key and is used in conjunction with a secret access key to sign requests cryptographically.
- SK: secret access key used in conjunction with an AK to sign requests cryptographically. It identifies a request sender and prevents the request from being modified.

In authentication using AK/SK, you can use AK/SK to sign requests based on the signature algorithm or use a dedicated signature SDK to sign the requests.

For details about how to sign requests and use the signature SDK, see **API Signing Guide**.

NOTE

The signature SDK only supports signature, which is different from the SDKs provided by services.

3.3 Returned Values

Status Code

After sending a request, you will receive a response, including the status code, response header, and response body.

A status code is a group of digits ranging from 1xx to 5xx. It indicates the status of a response. For more information, see **Status Codes**.

For example, if status code **201** is returned for calling the API used to **obtain a user token**, the request is successful.

Response Header

A response header corresponds to a request header, for example, **Content-Type**.

Figure **Figure 3-1** shows the response header fields for the API that obtains a user token. The **x-subject-token** value is the desired user token. You can use the token to authenticate other API calls.

Figure 3-1 Header of the response to the request for obtaining a user token

Response Body

A response body is generally returned in a structured format, corresponding to the **Content-Type** in the response header, and is used to transfer content other than the response header.

The following shows part of the response body for the API to **obtain a user token**. The following describes only part of the request body.

```
{
    "token": {
        "expires_at": "2019-02-13T06:52:13.855000Z",
        "methods": [
            "password"
        ],
        "catalog": [
            {
                "endpoints": [
                    {
                        "region_id": "az-01",
.....
```

If an error occurs during API calling, the system returns an error code and a message to you. The following shows the format of an error response body:

```
{
    "error_msg": "The format of message is error",
    "error_code": "AS.0001"
}
```


In the preceding information, **error_code** is an error code, and **error_msg** describes the error.

4 Getting Started

4.1 Creating a Queue

Scenario Description

This section describes how to create and query a queue using APIs.

Constraints

- Queues created using this API will be bound to specified compute resources.
- It takes 6 to 10 minutes to start a job using a new queue for the first time.

Involved APIs

- **Creating a Queue**: Create a queue.
- **Viewing Details of a Queue**: Ensure that the queue is successfully created.

Procedure

1. Create a queue.
   - API

     URI format: POST /v1.0/{project_id}/queues
     - Obtain the value of {project_id} from **Obtaining a Project ID**.
     - For details about the request parameters, see **Creating a Queue**.
   - Request example
     - Description: Create an SQL queue named **queue1** in the project whose ID is **48cc2c48765f481480c7db940d6409d1**.
     - Example URL: POST https://{*endpoint*}/v1.0/48cc2c48765f481480c7db940d6409d1/queues
     - Body:

       ```
       {
         "queue_name": "queue1",
       ```



```
    "description": "test",
    "cu_count": 16,
    "resource_mode": 1,
    "queue_type": "sql"
}
```

- Example response

```
{
  "is_success": true,
  "message": "",
  "queue_name": "queue1"
}
```

2. Verify that the queue is created successfully.
   - API

     URI format: GET /v1.0/{*project_id*}/queues/{*queue_name*}

     - Obtain the value of {*project_id*} from **Obtaining a Project ID**.
     - For details about the query parameters, see **Viewing Details of a Queue**.
   - Request example
     - Description: Query details about queue1 in the project whose ID is 48cc2c48765f481480c7db940d6409d1.
     - Example URL: GET https://{*endpoint*}/v1.0/48cc2c48765f481480c7db940d6409d1/queues/queue1
     - Body:

       ```
       {}
       ```
   - Example response

```
{
    "is_success": true,
    "message": "",
    "owner": "testuser",
    "description": "",
    "queue_name": "queue1",
    "create_time": 1587613028851,
    "queue_type": "sql",
    "cu_count": 16,
    "resource_id": "03d51b88-db63-4611-b779-9a72ba0cf58b",
    "resource_mode": 0
}
```

4.2 Creating and Submitting a SQL Job

Scenario Description

This section describes how to create and query SQL jobs using APIs.

Constraints

- It takes 6 to 10 minutes to start a job using a new queue for the first time.

Involved APIs

- **Creating a Queue**: Create a queue.

- **Creating a Database (Discarded)**: Create a database.
- **Creating a Table (Discarded)**: Create a table.
- **Importing Data (Discarded)**: Import the data to be queried.
- **Querying Job Details**: Check whether the imported data is correct.
- **Submitting a SQL Job (Recommended)**: Submit a query job.

Procedure

1. Create a SQL queue. For details, see **Creating a Queue**.
2. Create a database.
   - API

     URI format: POST /v1.0/{project_id}/databases

     - Obtain the value of {project_id} from **Obtaining a Project ID**.
     - For details about the request parameters, see **Creating a Database (Discarded)**.
   - Request example
     - Description: Creates a database named **db1** in the project whose ID is **48cc2c48765f481480c7db940d6409d1**.
     - Example URL: POST https://{*endpoint*}/v1.0/48cc2c48765f481480c7db940d6409d1/databases
     - Body:

       ```
       {
           "database_name": "db1",
           "description": "this is for test"
       }
       ```
   - Example response

     ```
     {
       "is_success": true,
       "message": ""
     }
     ```
3. Create a table.
   - API

     URI format: POST /v1.0/{*project_id*}/databases/{*database_name*}/tables

     - Obtain the value of {*project_id*} from **Obtaining a Project ID**.
     - For details about the request parameters, see **Creating a Table (Discarded)**.
   - Request example
     - Description: In the project whose ID is **48cc2c48765f481480c7db940d6409d1**, create a table named **tb1** in the **db1** database.
     - Example URL: POST https://{*endpoint*}/v1.0/48cc2c48765f481480c7db940d6409d1/databases/db1/tables
     - Body:


```
{
  "table_name": "tb1",
  "data_location": "OBS",
  "description": "",
  "data_type": "csv",
  "data_path": "obs://obs/path1/test.csv",
  "columns": [
  {
     "column_name": "column1",
     "type": "string",
     "description": "",
     "is_partition_column": true
  },
  {
     "column_name": "column2",
     "type": "string",
     "description": "",
     "is_partition_column": false
  }
  ],
  "with_column_header": true,
  "delimiter": ",",
  "quote_char": "\"",
  "escape_char": "\\",
  "date_format": "yyyy-MM-dd",
  "timestamp_format": "yyyy-MM-dd HH:mm:ss"
}
```

- Example response

```
{
  "is_success": true,
  "message": ""
}
```

4. (Optional) If the table to be created does not contain data, use the **Importing Data (Discarded)** API to import data to the table.
5. (Optional) After data is imported, you can use the **Querying Job Details** API to check whether the imported data is correct.
6. Submit a query job.
   - API

     URI format: POST /v1.0/{*project_id*}/jobs/submit-job

     - Obtain the value of {*project_id*} from **Obtaining a Project ID**.
     - For details about the request parameters, see **Creating a Database (Discarded)**.
   - Request example
     - Description: Submit a SQL job in the project whose ID is **48cc2c48765f481480c7db940d6409d1** and query data in the **tb1** table in the database **db1**.
     - Example URL: POST https://{*endpoint*}/v1.0/48cc2c48765f481480c7db940d6409d1/jobs/submit-job
     - Body:

```
{
   "currentdb": "db1",
   "sql": "select * from tb1 limit 10",
   "queue_name": "queue1"
}
```

   - Example response


```
{
  "is_success": true,
  "message": "",
  "job_id":""95fcc908-9f1b-446c-8643-5653891d9fd9",
  "job_type": "QUERY",
  "job_mode": "async"
}
```

4.3 Creating and Submitting a Spark Job

Scenario Description

This section describes how to create and submit Spark jobs using APIs.

Constraints

- It takes 6 to 10 minutes to start a job using a new queue for the first time.

Involved APIs

- **Creating a Queue**: Create a queue.
- **Uploading a Package Group (Discarded)**: Upload the resource package required by the Spark job.
- **Querying Resource Packages in a Group (Discarded)**: Check whether the uploaded resource package is correct.
- **Creating a Batch Processing Job**: Create and submit a Spark batch processing job.
- **Querying a Batch Job Status**: View the status of a batch processing job.
- **Querying Batch Job Logs (Discarded)**: View batch processing job logs.

Procedure

1. Create a common queue. For details, see **Creating a Queue**.
2. Upload a package group.
   - API

     URI format: POST /v2.0/{*project_id*}/resources

     - Obtain the value of {*project_id*} from **Obtaining a Project ID**.
     - For details about the request parameters, see **Uploading a Package Group (Discarded)**.
   - Request example
     - Description: Upload resources in the GATK group to the project whose ID is **48cc2c48765f481480c7db940d6409d1**.
     - Example URL: POST https://{*endpoint*}/v2.0/48cc2c48765f481480c7db940d6409d1/resources
     - Body:

       ```
       {
          "paths": [
             "https://test.obs.xxx.com/txr_test/jars/spark-sdv-app.jar"
          ],
       ```



```
    "kind": "jar",
    "group": "gatk",
    "is_async":"true"
}
```

- Example response

```
{
  "group_name": "gatk",
  "status": "READY",
  "resources": [
    "spark-sdv-app.jar",
    "wordcount",
    "wordcount.py"
  ],
  "details": [
    {
      "create_time": 0,
      "update_time": 0,
      "resource_type": "jar",
      "resource_name": "spark-sdv-app.jar",
      "status": "READY",
      "underlying_name": "987e208d-d46e-4475-a8c0-a62f0275750b_spark-sdv-app.jar"
    },
    {
      "create_time": 0,
      "update_time": 0,
      "resource_type": "jar",
      "resource_name": "wordcount",
      "status": "READY",
      "underlying_name": "987e208d-d46e-4475-a8c0-a62f0275750b_wordcount"
    },
    {
      "create_time": 0,
      "update_time": 0,
      "resource_type": "jar",
      "resource_name": "wordcount.py",
      "status": "READY",
      "underlying_name": "987e208d-d46e-4475-a8c0-a62f0275750b_wordcount.py"
    }
  ],
  "create_time": 1551334579654,
  "update_time": 1551345369070
}
```

3. View resource packages in a group.
   - API

     URI format: GET /v2.0/{*project_id*}/resources/{*resource_name*}

     - Obtain the value of {project_id} from **Obtaining a Project ID**.
     - For details about the query parameters, see **Creating a Table (Discarded)**.
   - Request example
     - Description: Query the resource package named **luxor-router-1.1.1.jar** in the GATK group under the project whose ID is **48cc2c48765f481480c7db940d6409d1**.
     - Example URL: GET https://{*endpoint*}/v2.0/48cc2c48765f481480c7db940d6409d1/resources/luxor-router-1.1.1.jar?group=gatk
     - Body:

       ```
       {}
       ```


- Example response

```
{
    "create_time": 1522055409139,
    "update_time": 1522228350501,
    "resource_type": "jar",
    "resource_name": "luxor-router-1.1.1.jar",
    "status": "uploading",
    "underlying_name": "7885d26e-c532-40f3-a755-c82c442f19b8_luxor-router-1.1.1.jar",
    "owner": "****"
}
```

4. Create and submit a Spark batch processing job.
   - API

     URI format: POST /v2.0/{*project_id*}/batches

     - Obtain the value of {*project_id*} from **Obtaining a Project ID**.
     - For details about the request parameters, see **Creating a Batch Processing Job**.
   - Request example
     - Description: In the **48cc2c48765f481480c7db940d6409d1** project, create a batch processing job named **TestDemo4** in **queue1**.
     - Example URL: POST https://{*endpoint*}/v2.0/48cc2c48765f481480c7db940d6409d1/batches
     - Body:

```
{
  "sc_type": "A",
  "jars": [

"spark-examples_2.11-2.1.0.luxor.jar"
  ],
  "driverMemory": "1G",
  "driverCores": 1,
  "executorMemory": "1G",
  "executorCores": 1,
  "numExecutors": 1,
  "queue": "cce_general",
  "file":
"spark-examples_2.11-2.1.0.luxor.jar",
  "className":
"org.apache.spark.examples.SparkPi",
  "minRecoveryDelayTime": 10000,
  "maxRetryTimes": 20
}
```

   - Example response

```
{
  "id": "07a3e4e6-9a28-4e92-8d3f-9c538621a166",
  "appId": "",
  "name": "",
  "owner": "test1",
  "proxyUser": "",
  "state": "starting",
  "kind": "",
  "log": [],
  "sc_type": "CUSTOMIZED",
  "cluster_name": "aaa",
  "queue": "aaa",
  "create_time": 1607589874156,
  "update_time": 1607589874156
}
```


5. Query a batch job status.
   - API

     URI format: GET /v2.0/{*project_id*}/batches/{*batch_id*}/state

     - Obtain the value of {*project_id*} from **Obtaining a Project ID**.
     - For details about the query parameters, see **Querying a Batch Job Status**.
   - Request example
     - Description: Query the status of the batch processing job whose ID is **0a324461-d9d9-45da-a52a-3b3c7a3d809e** in the project whose ID is **48cc2c48765f481480c7db940d6409d1**.
     - Example URL: GET https://{*endpoint*}/v2.0/48cc2c48765f481480c7db940d6409d1/batches/0a324461-d9d9-45da-a52a-3b3c7a3d809e/state
     - Body:
       ```
       {}
       ```
   - Example response
     ```
     {
       "id":"0a324461-d9d9-45da-a52a-3b3c7a3d809e",
       "state":"Success"
     }
     ```
6. Query batch job logs.
   - API

     URI format: GET /v2.0/{*project_id*}/batches/{*batch_id*}/log

     - Obtain the value of {*project_id*} from **Obtaining a Project ID**.
     - For details about the query parameters, see **Querying Batch Job Logs (Discarded)**.
   - Request example
     - Description: Query the background logs of the batch processing job **0a324461-d9d9-45da-a52a-3b3c7a3d809e** in the **48cc2c48765f481480c7db940d6409d1** project.
     - Example URL: GET https://{*endpoint*}/v2.0/48cc2c48765f481480c7db940d6409d1/batches/0a324461-d9d9-45da-a52a-3b3c7a3d809e/log
     - Body:
       ```
       {}
       ```
   - Example response
     ```
     {
       "id": "0a324461-d9d9-45da-a52a-3b3c7a3d809e",
       "from": 0,
       "total": 3,
       "log": [
           "Detailed information about job logs"
       ]
     }
     ```


4.4 Creating and Submitting a Flink Job

Scenario Description

This section describes how to create and run a user-defined Flink job using APIs.

Constraints

- It takes 6 to 10 minutes to start a job using a new queue for the first time.

Involved APIs

- **Creating a Queue**: Create a queue.
- **Uploading a Package Group (Discarded)**: Upload the resource package required by the Flink custom job.
- **Querying Resource Packages in a Group (Discarded)**: Check whether the uploaded resource package is correct.
- **Creating a Flink Jar job** Create a user-defined Flink job.
- **Running Jobs in Batches**: Run a user-defined Flink job.

Procedure

1. Create a queue for general use. For details, see **Creating a Queue**. In the request, set **resource_mode** to **1** to create a dedicated queue.
2. Upload the resource package of the user-defined Flink job. For details, see **2**.
3. Query resource packages in a group. For details, see **3**.
4. Create a custom flink job.
   - API

     URI format: POST /v1.0/{*project_id*}/streaming/flink-jobs

     - Obtain the value of {*project_id*} from **Obtaining a Project ID**.
     - For details about the request parameters, see **Creating a Database (Discarded)**.
   - Request example
     - Description: Create a user-defined Flink job in the project whose ID is **48cc2c48765f481480c7db940d6409d1**.
     - Example URL: POST https://{*endpoint*}/v1.0/48cc2c48765f481480c7db940d6409d1/streaming/flink-jobs
     - Body:

```
{
    "name": "test",
    "desc": "job for test",
    "queue_name": "testQueue",
    "manager_cu_number": 1,
    "cu_number": 2,
    "parallel_number": 1,
    "tm_cus": 1,
    "tm_slot_num": 1,
```



```
    "log_enabled": true,
    "obs_bucket": "bucketName",
    "smn_topic": "topic",
    "main_class": "org.apache.flink.examples.streaming.JavaQueueStream",
    "restart_when_exception": false,
    "entrypoint": "javaQueueStream.jar",
    "entrypoint_args":"-windowSize 2000 -rate3",
    "dependency_jars": [
        "myGroup/test.jar",
        "myGroup/test1.jar"
    ],
    "dependency_files": [
        "myGroup/test.csv",
        "myGroup/test1.csv"
    ]
}
```

- Example response

```
{
  "is_success": true,
  "message": "A Flink job is created successfully.",
  "job": {
    "job_id": 138,
    "status_name": "job_init",
    "status_desc": ""
  }
}
```

5. Run jobs in batches.
   - API

     URI format: POST /v1.0/{*project_id*}/streaming/jobs/run

     - Obtain the value of {*project_id*} from **Obtaining a Project ID**.
     - For details about the request parameters, see **Running Jobs in Batches**.
   - Request example
     - Description: Run the jobs whose **job_id** is **298765** and **298766** in the project whose ID is **48cc2c48765f481480c7db940d6409d1**.
     - Example URL: POST https://{*endpoint*}/v1.0/48cc2c48765f481480c7db940d6409d1/streaming/jobs/run
     - Body:

       ```
       {
           "job_ids": [131,130,138,137],
           "resume_savepoint": true
       }
       ```
   - Example response

     ```
     [
       {
           "is_success": "true",
           "message": "The request for submitting DLI jobs is delivered successfully."
       },
       {
           "is_success": "true",
           "message": "The request for submitting DLI jobs is delivered successfully."
       },
       {
           "is_success": "true",
           "message": "The request for submitting DLI jobs is delivered successfully."
       },
       {
           "is_success": "true",
     ```



```
    "message": "The request for submitting DLI jobs is delivered successfully."
  }
]
```

4.5 Creating and Using a Datasource Connection

Scenario Description

This section describes how to create an enhanced datasource connection using an API.

Constraints

- It takes 6 to 10 minutes to start a job using a new queue for the first time.
- Before creating an enhanced datasource connection, you need to obtain the ID of the VPC and the network ID of the subnet where the service is located.

Involved APIs

- **Creating a Queue**: Create a dedicated queue.
- **Creating an Enhanced Datasource Connection**: Create an enhanced datasource connection.
- **Binding a Queue**: Bind a queue.
- **Querying an Enhanced Datasource Connection**: Check whether an enhanced datasource connection is successfully created.

Procedure

1. Create a queue. For details, see **Creating a Queue**. In the request, set **resource_mode** to **1** to create a dedicated queue.
2. Create an enhanced datasource connection.
   - API

     URI format: POST /v2.0/{*project_id*}/datasource/enhanced-connections

     - Obtain the value of {*project_id*} from **Obtaining a Project ID**.
     - For details about the request parameters, see **Creating an Enhanced Datasource Connection**.
   - Request example
     - Description: Create an enhanced datasource connection named **test1** in project **48cc2c48765f481480c7db940d6409d1**.
     - Example URL: POST https://{*endpoint*}/v2.0/48cc2c48765f481480c7db940d6409d1/datasource/enhanced-connections
     - Body:

```
{
  "name": "test1",
  "dest_vpc_id": "22094d8f-c310-4621-913d-4c4d655d8495",
  "dest_network_id": "78f2562a-36e4-4b39-95b9-f5aab22e1281",
  "queues": ["q1","q2"],
  "hosts": [
```



```
    {
      "ip":"192.168.0.1",
      "name":"ecs-97f8-0001"
    },
    {
      "ip":"192.168.0.2",
      "name":"ecs-97f8-0002"
    }
  ]
}
```

- Example response

```
{
  "is_success": true,
  "message": "",
  "connection_id": "2a620c33-5609-40c9-affd-2b6453071b0f"
}
```

3. (Optional) If no queue is bound when you create an enhanced datasource connection, you can use the **Binding a Queue** API to bind a queue.
4. Verify that the enhanced datasource connection is created successfully.
   - API

     URI format: GET /v2.0/{*project_id*}/datasource/enhanced-connections/{*connection_id*}

     - Obtain the value of {*project_id*} from **Obtaining a Project ID**.
     - For details about the query parameters, see **Creating a Database (Discarded)**.
   - Request example
     - Description: Query an enhanced datasource connection whose ID is **2a620c33-5609-40c9-affd-2b6453071b0f** in project **48cc2c48765f481480c7db940d6409d1**.
     - Example URL: GET https://{*endpoint*}/v2.0/48cc2c48765f481480c7db940d6409d1/datasource/enhanced-connections/2a620c33-5609-40c9-affd-2b6453071b0f
     - Body:

       ```
       {}
       ```
   - Example response

     ```
     {
       "is_success": true,
       "message": "",
       "name": "test1",
       "id": "2a620c33-5609-40c9-affd-2b6453071b0f",
       "available_queue_info": [
         {
           "status": "ACTIVE",
           "name": "queue1",
           "peer_id": "2a620c33-5609-40c9-affd-2b6453071b0f",
           "err_msg": "",
           "update_time": 1566889577861
         }
       ],
       "dest_vpc_id": "22094d8f-c310-4621-913d-4c4d655d8495",
       "dest_network_id": "78f2562a-36e4-4b39-95b9-f5aab22e1281",
       "isPrivis": true,
       "create_time": 1566888011125,
       "status": "ACTIVE",
       "hosts": [
         {
     ```



```
    "ip":"192.168.0.1",
    "name":"ecs-97f8-0001"
   },
   {
    "ip":"192.168.0.2",
    "name":"ecs-97f8-0002"
   }
  ]
}
```


5 Permission-related APIs

5.1 Granting Data Access Control to Users or Projects

Function

This API is used to grant data permissions for a database or data table to a specified user or project.

The user group containing the authorized user must have the **Tenant Guest** permission in the region where the user group belongs.

Permissions can only be granted to projects within the same region of the same tenant.

URI

- URI format

  PUT /v1.0/{project_id}/authorization
- Parameter description

  Table 5-1 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 5-2 Request parameters

Parameter	Mandatory	Type	Description
user_name	No	String	Name of the user who is granted, revoked, or updated permissions on a database or data table
grant_project_id	No	String	ID of the project that is granted permissions to access a database or data table. Once granted, the project administrator will have access to the database or data table.
action	Yes	String	Grants, revokes, or updates permissions. • **grant**: Grants permissions. • **revoke**: Revokes permissions. • **update**: Updates permissions. **NOTE** Users can perform the **update** operation only when they have been granted the **grant** and **revoke** permissions.
privileges	Yes	Array of objects	Granted permissions. For details, see **Table 5-3**.

Table 5-3 privileges parameters

Parameter	Mandatory	Type	Description
object	Yes	String	Data objects whose permissions are to be granted. If they are named: • **databases.***Database name*, all data in the database will be shared. • **databases.***Database name*.**tables.***Table name*, data in a specified table will be shared. • **databases.***Database name*.**tables.***Table name*.**columns.***Column name*, data in a specified column will be shared. • **edsconnections.***Connection ID*: grants the permission to use an enhanced datasource connection.

Parameter	Mandatory	Type	Description
privileges	Yes	Array of strings	List of permissions to be granted, revoked, or updated **NOTE** If **action** is set to **update** and the update list is empty, the user's all permissions on the database or data table are revoked.

Response

Table 5-4 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If the execution succeeds, the message may be left blank.

Example Request

- Grant a project (ID: **0732e57c728025922f04c01273686950**) the permission to query data in the database **db1**, delete the data table **db1.tbl**, and query data in a specified column **db1.tbl.column1** of a data table.

```
{
  "grant_project_id": "0732e57c728025922f04c01273686950",
  "action": "grant",
  "privileges": [
    {
      "object": "databases.db1.tables.tb2.columns.column1",
      "privileges": ["SELECT"]
    },
    {
      "object": "databases.db1.tables.tbl",
      "privileges": ["DROP_TABLE"]
    },
    {
  "object": "databases.db1",
  "privileges": ["SELECT"]
    }
  ]
}
```

- Grant the data query permission of the **dbtest** database to the **dlitest** user.

```
{
    "action": "grant",
    "privileges": [
        {
            "object": "databases.dbtest",
            "privileges": [
                "SELECT"
            ]
```



```
        }
    ],
    "user_name": "dlitest"
}
```

Example Response

```
{
  "is_success": true,
  "message": ""
}
```

Status Codes

Table 5-5 describes the status codes.

Table 5-5 Status codes

Status Code	Description
200	Authorization succeeds.
400	Request error.
500	Internal server error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

5.2 Checking the Permissions Granted to a User

Function

This API is used to check the permissions granted to a user.

URI

- URI format
  GET /v1.0/{project_id}/authorization/privileges
- Parameter descriptions:

Table 5-6 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 5-7 query parameter description

Parameter	Mandatory	Type	Description
object	Yes	String	Data object to be assigned, which corresponds to the **object** in API permission assignment. • **jobs.flink.*Fink job ID***, data in the specified job will be queried. • **groups. *Package group name***, data in the specified package group will be queried. • **resources.*Package name***, data in the specified package will be queried. **NOTE** When you view the packages in a group, the **object** format is **resources.package group name/package name**.

NOTE

The following is an example of the URL containing the **query** parameter:
GET /v1.0/{project_id}/authorization/privileges*?object={object}*

Request

None

Response

Table 5-8 Response parameters

Parameter	Mandatory	Type	Description
is_success	Yes	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	Yes	String	Indicates the system prompt. If execution succeeds, this parameter may be left blank.
object_name	No	String	Object name.
object_type	No	String	Object type.
privileges	No	Array of Object	Permission information. For details, see **Table 5-9**.

Parameter	Mandatory	Type	Description
count	No	Integer	Total number of permissions.

Table 5-9 privileges parameters

Parameter	Mandatory	Type	Description
object	No	String	Data object whose permissions are to be granted, which corresponds to the **object** in the permission assignment API.
is_admin	No	Boolean	Whether the database user is an administrator.
user_name	No	String	Name of the user who has permission on the current database.
privileges	No	Array of Strings	Permission of the user on the database.

Example Request

None

Example Response

```
{
    "is_success": true,
    "message": "",
    "object_name": "9561",
    "object_type": "flink",
    "count": 2,
    "privileges": [
        {
            "user_name": "testuser1",
            "is_admin": true,
            "privileges": [
                "ALL"
            ]
        },
        {
            "user_name": "user1",
            "is_admin": false,
            "privileges": [
                "GET"
            ]
        }
    ]
}
```

Status Codes

Table 5-10 describes the status code.

Table 5-10 Status codes

Status Code	Description
200	Authorization succeeds.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 5-11 Error codes

Error Code	Error Message
DLI.0001	user input validation failed, object_type sql or saprk is not supported now

6 Global Variable-related APIs

6.1 Creating a Global Variable

Function

This API is used to create a global variable.

URI

- URI format
  POST /v1.0/{project_id}/variables
- Parameter description

Table 6-1 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 6-2 Request parameters

Parameter	Mandatory	Type	Description
var_name	Yes	String	A global variable name can contain a maximum of 128 characters, including only digits, letters, and underscores (_), but cannot start with an underscore (_) or contain only digits.
var_value	Yes	String	Global variable value.
is_sensitive	No	Boolean	Whether to set a variable as a sensitive variable. The default value is **false**.

Response

Table 6-3 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	Message content.

Example Request

Create a global variable that is sensitive.

```
{
    "var_name": "string",
    "var_value": "string",
    "is_sensitive": true
}
```

Example Response

```
{
    "is_success": true,
    "message": "string"
}
```

Status Codes

Table 6-4 describes status codes.

Table 6-4 Status codes

Status Code	Description
200	A variable is created successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 6-5 Error codes

Error Code	Error Message
DLI.0001	Parameter check errors occur.
DLI.0999	The object exists.

6.2 Deleting a Global Variable

Function

This API is used to delete a global variable.

NOTE

Only the user who creates a global variable can delete the variable.

URI

- URI format

  DELETE /v1.0/{project_id}/variables/{var_name}
- Parameter description

Table 6-6 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Parameter	Mandatory	Type	Description
var_name	Yes	String	A global variable name can contain a maximum of 128 characters, including only digits, letters, and underscores (_), but cannot start with an underscore (_) or contain only digits.

Request

None

Response

- Parameter description

Table 6-7 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Example Request

None

Example Response

```
{
    "is_success": true,
    "message": "string"
}
```

Status Codes

Table 6-8 describes status codes.

Table 6-8 Status codes

Status Code	Description
200	A variable is deleted successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 6-9 Error codes

Error Code	Error Message
DLI.0001	Parameter check errors occur.
DLI.0999	Server-side errors occur.

6.3 Modifying a Global Variable

Function

This API is used to modify a global variable.

URI

- URI format

  PUT /v1.0/{project_id}/variables/{var_name}
- Parameter description

  Table 6-10 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
var_name	Yes	String	A global variable name can contain a maximum of 128 characters, including only digits, letters, and underscores (_), but cannot start with an underscore (_) or contain only digits.

Request

Table 6-11 Request parameters

Parameter	Mandatory	Type	Description
var_value	Yes	String	Global variable value.

Response

Table 6-12 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	Message content.

Example Request

Change the value of a sensitive variable.

```
{
    "var_value": "string"
}
```

Example Response

```
{
    "is_success": true,
    "message": "string"
}
```

Status Codes

Table 6-13 describes status codes.

Table 6-13 Status codes

Status Code	Description
200	A variable is modified successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 6-14 Error codes

Error Code	Error Message
DLI.0001	Parameter check errors occur.
DLI.0999	Server-side errors occur.
DLI.12004	The job does not exist. Check the reason or create a job.

6.4 Querying All Global Variables

Function

This API is used to query information about all global variables in the current project.

URI

- URI format
  GET /v1.0/{project_id}/variables
- Parameter description

Table 6-15 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 6-16 query parameter description

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of returned records displayed on each page. The default value is **100**.
offset	No	Integer	Offset. The default value is **0**.

Request

None

Response

Table 6-17 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
count	No	Integer	Number of global variables.
global_vars	No	Array of objects	Global variable information. For details, see **Table 6-18**.

Table 6-18 global_vars parameters

Parameter	Mandatory	Type	Description
id	No	Long	Global variable ID.
var_name	Yes	String	Global variable name.
var_value	Yes	String	Global variable value.
project_id	No	String	Project ID.
user_id	No	String	User ID.
user_name	No	String	Username
is_sensitive	No	Boolean	Whether to set a variable as a sensitive variable.
create_time	No	Long	Creation time
update_time	No	Long	Update time

Example Request

None

Example Response

```
{
    "is_success": true,
```

```
  "message": "string",
  "count": 0,
  "global_vars": [
    {
      "id": 0,
      "var_name": "string",
      "var_value": "string",
      "project_id": "string",
      "user_id": "string"
    }
  ]
}
```


Status Codes

Table 6-19 Status codes

Status Code	Description
200	All variables are queried successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 6-20 Error codes

Error Code	Error Message
DLI.0001	Parameter check errors occur.
DLI.0999	Server-side errors occur.

7 APIs Related to Enhanced Datasource Connections

7.1 Creating an Enhanced Datasource Connection

Function

This API is used to create an enhanced datasource connection with other services.

NOTE

- For details about how to use the console interface, see **Enhanced Datasource Connections** in *Data Lake Insight User Guide*.
- Datasource connections cannot be created for the default queue.

URI

- URI format
  POST /v2.0/{project_id}/datasource/enhanced-connections
- Parameter description

Table 7-1 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 7-2 Request parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the connection. • The name can contain only letters, digits, and underscores (_), and cannot be left empty. • A maximum of 64 characters are allowed.
dest_vpc_id	Yes	String	The ID of the service VPC to be connected.
dest_network_id	Yes	String	The subnet ID of the to-be-connected service.
elastic_resource_pools	No	Array of Strings	List of elastic resource pools that need to use datasource resources.
queues	No	Array of Strings	This parameter is about to be taken offline with DLI queues. Use the **elastic_resource_pools** parameter instead. List of queue names that are available for datasource connections. **NOTE** Only dedicated queues are allowed to use the enhanced datasource connections.
routetable_id	No	String	Route table associated with the subnet of the service.
hosts	No	Array of objects	The user-defined host information. A maximum of 20,000 records are supported. For details, see **hosts request parameters**.
tags	No	Array of objects	Tags of datasource connections. For details, see **Table 7-4**.

Table 7-3 hosts request parameters

Parameter	Mandatory	Type	Description
name	No	String	The user-defined host name. The value can consist of 128 characters, including digits, letters, underscores (_), hyphens (-), and periods (.). It must start with a letter.
ip	No	String	The IPv4 address of the host.

Table 7-4 tags parameter

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key. **NOTE** A tag key can contain a maximum of 36 characters. Special characters (=*<>|) are not allowed, and the key cannot start with a space.
value	Yes	String	Tag key. **NOTE** A tag value can contain a maximum of 43 characters. Special characters (=*<>|) are not allowed, and the value cannot start with a space.

Response

Table 7-5 Response parameters

Parameter	Type	Description
is_success	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	String	System prompt. If execution succeeds, the message may be left blank.
connection_id	String	Connection ID. Identifies the UUID of a datasource connection.

Example Request

Create an enhanced datasource connection for a queue and configure host information.

```
{
  "name": "test",
```



```
 "dest_vpc_id": "22094d8f-c310-4621-913d-4c4d655d8495",
 "dest_network_id": "78f2562a-36e4-4b39-95b9-f5aab22e1281",
 "elastic_resource_pools": [
  "e1"
 ],
 "hosts": [
  {
   "ip":"192.168.0.1",
   "name":"ecs-97f8-0001"
  },
  {
   "ip":"192.168.0.2",
   "name":"ecs-97f8-0002"
  }
 ]
}
```

Example Response

```
{
 "is_success": true,
 "message": "Create peer connection for queues:{Resource pool list in the request parameter}",
 "connection_id": "2a620c33-5609-40c9-affd-2b6453071b0f"
}
```

Status Codes

Table 7-6 describes the status code.

Table 7-6 Status codes

Status Code	Description
201	The job is created successfully.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

7.2 Deleting an Enhanced Datasource Connection

Function

This API is used to delete an enhanced datasource connection.

NOTE

The connection that is being created cannot be deleted.

URI

- URI format

  DELETE /v2.0/{project_id}/datasource/enhanced-connections/{connection_id}
- Parameter description

  Table 7-7 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
connection_id	Yes	String	Connection ID. Identifies the UUID of a datasource connection. Set the value to the connection ID returned by **Creating an Enhanced Datasource Connection**.

Request

None

Response

Table 7-8 Response parameters

Parameter	Type	Description
is_success	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	String	System message. Value **Deleted** indicates that the operation is successful.

Example Request

None

Example Response

```
{
    "is_success": true,
    "message": "Deleted"
}
```

Status Codes

Table 7-9 describes the status code.

Table 7-9 Status codes

Status Code	Description
200	Deletion succeeded.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

7.3 Listing Enhanced Datasource Connections

Function

This API is used to list the created enhanced datasource connections.

URI

- URI format

  GET /v2.0/{project_id}/datasource/enhanced-connections
- Parameter description

Table 7-10 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 7-11 query parameter description

Parameter	Mandatory	Type	Description
limit	No	String	The maximum number of connections to be queried. The default value is **100**. If **limit** is set to **0**, all datasource connections are returned.

Parameter	Mandatory	Type	Description
offset	No	String	The offset of the query result. The default value is **0**. Note that the connections are sorted by creation time.
status	No	String	Connection status. The options are as follows: • Active: The connection has been activated. • DELETED: The connection has been deleted. **NOTE** The connection status is case insensitive.
name	No	String	Connection name
tags	No	String	List of tag names. The value is **k=v** for a single tag. Multiple tags are separated by commas (,). Example: **tag1=v1,tag2=v2**.

NOTE

The following is an example of the URL containing the **query** parameter:

GET /v2.0/{project_id}/datasource/enhanced-connections?limit=*{limit}*&offset=*{offset}*&status=*{status}*&name=*{name}*

Request

None

Response

Table 7-12 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
connections	No	Array of objects	Datasource connection information list. For details, see **Table 7-13**.

Parameter	Mandatory	Type	Description
count	No	Integer	Number of returned datasource connections.

Table 7-13 connections parameters

Parameter	Mandatory	Type	Description
id	No	String	Connection ID. Identifies the UUID of a datasource connection.
name	No	String	User-defined connection name.
status	No	String	Connection status. The options are as follows: • Active: The connection has been activated. • DELETED: The connection has been deleted.
available_queue_info	No	Array of objects	For details about how to create a datasource connection for each queue, see **Table 7-14**.
dest_vpc_id	No	String	The VPC ID of the connected service.
dest_network_id	No	String	Subnet ID of the connected service.
isPrivis	No	Boolean	Whether the project permissions have been granted for the enhanced datasource connection. If the datasource connection has the permissions, the value of this field is **false**. Otherwise, the value is **true**.
create_time	No	Long	Time when a link is created. The time is converted to a UTC timestamp.
hosts	No	Array of objects	User-defined host information. For details, see **Table 7-15**.

Table 7-14 available_queue_info parameter description

Parameter	Mandatory	Type	Description
peer_id	No	String	ID of a datasource connection.
status	No	String	Connection status. For details about the status code, see **Table 7-16**.
name	No	String	Name of a queue.
err_msg	No	String	Detailed error message when the status is **FAILED**.
update_time	No	Long	Time when the available queue list was updated.

Table 7-15 hosts parameters

Parameter	Mandatory	Type	Description
name	No	String	Custom host name
ip	No	String	IPv4 address of the host

Table 7-16 Connection status

Parameter	Definition	Description
CREATING	Creating	The datasource connection is being created.
ACTIVE	Active	The datasource connection has been created, and the connection to the destination address is normal.
FAILED	Failed	Failed to create a datasource connection.

Example Request

None

Example Response

```
{
  "is_success": true,
  "message": "",
  "count": 1,
  "connections": [
    {
```



```
    "name": "withvpc",
    "id": "4c693ecc-bab8-4113-a838-129cedc9a563",
    "available_queue_info": [
      {
        "status": "ACTIVE",
        "name": "resource_mode_1",
        "peer_id": "d2ae6628-fa37-4e04-806d-c59c497492d1",
        "err_msg": "",
        "update_time": 1566889577861
      }
    ],
    "dest_vpc_id": "22094d8f-c310-4621-913d-4c4d655d8495",
    "dest_network_id": "78f2562a-36e4-4b39-95b9-f5aab22e1281",
    "isPrivis": true,
    "create_time": 1566888011125,
    "status": "ACTIVE"
  }
 ]
}
```


Status Codes

Table 7-17 describes the status code.

Table 7-17 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

7.4 Querying an Enhanced Datasource Connection

Function

This API is used to query a created enhanced datasource connection.

URI

- URI format
  GET /v2.0/{project_id}/datasource/enhanced-connections/{connection_id}
- Parameter description

Table 7-18 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
connection_id	Yes	String	Connection ID. Identifies the UUID of a datasource connection.

Request

None

Response

Table 7-19 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
id	No	String	Connection ID. Identifies the UUID of a datasource connection.
name	No	String	User-defined connection name.
status	No	String	Connection status. The options are as follows: • Active: The connection has been activated. • DELETED: The connection has been deleted.
available_queue_info	No	Array of objects	For details about how to create a datasource connection for each queue, see **Table 7-20**.
dest_vpc_id	No	String	The VPC ID of the connected service.
dest_network_id	No	String	Subnet ID of the connected service.

Parameter	Mandatory	Type	Description
create_time	No	Long	Time when a link is created. The time is converted to a UTC timestamp.
hosts	No	Array of objects	User-defined host information. For details, see **hosts parameter description**.

Table 7-20 available_queue_info parameter description

Parameter	Mandatory	Type	Description
peer_id	No	String	ID of a datasource connection.
status	No	String	Connection status. For details about the status code, see **Table 7-22**.
name	No	String	Name of a queue.
err_msg	No	String	Detailed error message when the status is **FAILED**.
update_time	No	Long	Time when the available queue list was updated.

Table 7-21 hosts parameter description

Parameter	Mandatory	Type	Description
name	No	String	The user-defined host name.
ip	No	String	The IPv4 address of the host.

Table 7-22 Connection status

Parameter	Definition	Description
CREATING	Creating	The datasource connection is being created.
ACTIVE	Active	The datasource connection has been created, and the connection to the destination address is normal.
FAILED	Failed	Failed to create a datasource connection.

Example Request

None

Example Response


```
{
  "is_success": true,
  "message": "",
  "name": "withvpc",
  "id": "4c693ecc-bab8-4113-a838-129cedc9a563",
  "available_queue_info": [
    {
      "status": "ACTIVE",
      "name": "resource_mode_1",
      "peer_id": "d2ae6628-fa37-4e04-806d-c59c497492d1",
      "err_msg": "",
      "update_time": 1566889577861
    }
  ],
  "dest_vpc_id": "22094d8f-c310-4621-913d-4c4d655d8495",
  "dest_network_id": "78f2562a-36e4-4b39-95b9-f5aab22e1281",
  "create_time": 1566888011125,
  "status": "ACTIVE",
  "hosts": [
    {
      "ip":"192.168.0.1",
      "name":"ecs-97f8-0001"
    },
    {
      "ip":"192.168.0.2",
      "name":"ecs-97f8-0002"
    }
  ]
}
```


Status Codes

Table 7-23 describes the status code.

Table 7-23 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

7.5 Binding a Queue

Function

This API is used to bind a queue to a created enhanced datasource connection.

URI

- URI format

  POST /v2.0/{project_id}/datasource/enhanced-connections/{connection_id}/associate-queue

- Parameter description

  Table 7-24 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
connection_id	Yes	String	Connection ID. Identifies the UUID of a datasource connection. Set the value to the connection ID returned by **Creating an Enhanced Datasource Connection**.

Request

Table 7-25 Request parameters

Parameter	Mandatory	Type	Description
queues	No	Array of Strings	List of queue names that are available for datasource connections.

Response

Table 7-26 Response parameters

Parameter	Type	Description
is_success	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.

Parameter	Type	Description
message	String	System prompt. If execution succeeds, the message may be left blank.

Example Request

Bind created enhanced datasource connections to queues **q1** and **q2**.

```
{
  "queues": [
    "q1",
    "q2"
  ]
}
```

Example Response

```
{
  "is_success": true,
  "message": "associated peer connection for queues: {q1,q2}."
}
```

Status Codes

Table 7-27 describes the status code.

Table 7-27 Status codes

Status Code	Description
200	Resource bound succeeded.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

7.6 Unbinding a Queue

Function

This API is used to unbind a queue from an enhanced datasource connection.

URI

- URI format

  POST /v2.0/{project_id}/datasource/enhanced-connections/{connection_id}/disassociate-queue

- Parameter description

  Table 7-28 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
connection_id	Yes	String	Connection ID. Identifies the UUID of a datasource connection.

Request

Table 7-29 Request parameters

Parameter	Mandatory	Type	Description
queues	No	Array of String	List of queue names that are available for datasource connections.

Response

Table 7-30 Response parameters

Parameter	Type	Description
is_success	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	String	System prompt. If execution succeeds, the message may be left blank.

Example Request

Unbind queues **q1** and **q2** from enhanced datasource connections.

```
{
  "queues": [
```



```
    "q1",
    "q2"
  ]
}
```

Example Response

```
{
  "is_success": true,
  "message": "Disassociated peer connection for queues:{q1,q2}."
}
```

Status Codes

[Table 7-31](#) describes the status code.

Table 7-31 Status codes

Status Codes	Description
200	Resource unbound succeeded.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

7.7 Modifying Host Information

Function

This API is used to modify the host information of a connected datasource. Only full overwriting is supported.

URI

- URI format

  PUT /v2.0/{project_id}/datasource/enhanced-connections/{connection_id}
- Parameter description

Table 7-32 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
connection_id	Yes	String	Connection ID. Identifies the UUID of a datasource connection.

Request

Table 7-33 Request parameters

Parameter	Mandatory	Type	Description
hosts	Yes	Array of objects	The user-defined host information. A maximum of 20,000 records are supported. For details, see **hosts request parameters**. If this parameter is left blank, all configured host information will be deleted.

Table 7-34 hosts request parameters

Parameter	Mandatory	Type	Description
name	No	String	The user-defined host name. The value can consist of 128 characters, including digits, letters, underscores (_), hyphens (-), and periods (.). It must start with a letter.
ip	No	String	The IPv4 address of the host.

Response

Table 7-35 Response parameters

Parameter	Type	Description
is_success	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.

Parameter	Type	Description
message	String	System prompt. If execution succeeds, the message may be left blank.

Example Request

Modify the host information of an enhanced datasource connection.

```
{
  "hosts": [
    {
      "ip":"192.168.0.1",
      "name":"ecs-97f8-0001"
    },
    {
      "ip":"192.168.0.2",
      "name":"ecs-97f8-0002"
    }
  ]
}
```

Example Response

```
{
  "is_success": true,
  "message": ""
}
```

Status Codes

Table 7-36 describes the status code.

Table 7-36 Status codes

Status Code	Description
200	The modification operations are successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

7.8 Querying Authorization of an Enhanced Datasource Connection

Function

This API is used to query the authorization about an enhanced datasource connection.

URI

- URI format

  GET /v2.0/{project_id}/datasource/enhanced-connections/{connection_id}/privileges

- Parameter description

  Table 7-37 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
connection_id	Yes	String	Connection ID. Identifies the UUID of a datasource connection.

Request

None

Response

Table 7-38 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
connection_id	No	String	Enhanced datasource connection ID, which is used to identify the UUID of a datasource connection.

Parameter	Mandatory	Type	Description
privileges	No	Array of Object	Datasource connection information about each authorized project. For details, see **Table 7-39**.

Table 7-39 privileges parameters

Parameter	Mandatory	Type	Description
object	No	String	Object information during authorization.
applicant_project_id	No	String	ID of an authorized project.
privileges	No	Array of Strings	Authorization operation information.

Example Request

None

Example Response

```
{
    "is_success": true,
    "message": "",
    "privileges": [
        {
            "object": "edsconnections.503fc86a-5e60-4349-92c2-7e399404fa8a",
            "applicant_project_id": "330e068af1334c9782f4226acc00a2e2",
            "privileges": ["BIND_QUEUE"]
        }
    ],
    "connection_id": "503fc86a-5e60-4349-92c2-7e399404fa8a"
}
```

Status Codes

Table 7-40 describes status codes.

Table 7-40 Status codes

Status Code	Description
200	The query is successful.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 7-41 Error codes

Error Code	Error Message
DLI.0001	Connection 503fc86a-5e60-4349-92c2-7e399404fa8a does not exist.

8 APIs Related to Elastic Resource Pools

8.1 Creating an Elastic Resource Pool

Function

This API is used to create elastic resource pools.

URI

- URI format
  POST /v3/{project_id}/elastic-resource-pools
- Parameter description

Table 8-1 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 8-2 Request parameters

Parameter	Mandatory	Type	Description
elastic_resource_pool_name	Yes	String	Name of a new elastic resource pool. Only digits, letters, and underscores (_) are allowed, but the value cannot contain only digits or start with an underscore (_). The value can contain 1 to 128 characters. **NOTE** If the name contains uppercase letters, the system automatically converts them to lowercase letters.
description	No	String	Description. The value can contain a maximum of 256 characters.
cidr_in_vpc	No	String	VPC CIDR associated with the virtual cluster. If it is not specified, the default value **172.16.0.0/12** is used.
max_cu	Yes	Integer	Maximum number of CUs. The value of this parameter must be greater than or equal to the sum of the maximum CUs allowed for any queue within the resource pool, and greater than the **min_cu** value. • Standard edition: The minimum value is 64 CUs. • Basic edition: The minimum value is 16 CUs, and the maximum value is 64 CUs.

Parameter	Mandatory	Type	Description
min_cu	Yes	Integer	Minimum number of CUs. The value of this parameter must be greater than or equal to the sum of the minimum CUs allowed for each queue in the resource pool. The minimum value is **64**. • Standard edition: The minimum value is 64 CUs. • Basic edition: The minimum value is 16 CUs, and the maximum value is 64 CUs.
charging_mode	No	Integer	Billing mode. The default value is **1**, which indicates the pay-per-use billing mode.
enterprise_project_id	No	String	Enterprise ID. If this parameter is left blank, the default value **0** is used.
tags	No	Array of objects	Queue tags for identifying cloud resources. A tag consists of a key and a value. For details, see **Table 8-3**. Currently, the tag key can only be **spec**, and the tag value can only be **basic**. **spec**: specifications type of the elastic resource pool to create. **basic**: a basic elastic resource pool. The basic edition allows for a value range of 16–64 CUs for setting **min_cu** and **max_cu**. If this parameter is left unspecified, a standard resource pool will be created.

Table 8-3 tags parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key
value	Yes	String	Tag value

Response

Table 8-4 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	Message content, for example, **Success to get tsdb list**.
elastic_resource_pool_name	No	String	Elastic resource pool name, for example, **elastic_pool_0623_02"**.

Example Request

Create an elastic resource pool with maximum CUs of 684 and minimum CUs of 684.

```
{
  "elastic_resource_pool_name" : "elastic_pool_0623_02",
  "description" : "test",
  "cidr_in_vpc" : "172.16.0.0/14",
  "charging_mode" : "1",
  "max_cu" : 684,
  "min_cu" : 684
}
```

Example Response

```
{
  "is_success" : true,
  "message" : "Success to get tsdb list",
  "elastic_resource_pool_name" : "elastic_pool_0623_02"
}
```

Status Codes

Status Code	Description
200	OK
400	Incorrect parameters. For example, creating an existing elastic resource pool.
403	Forbidden

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

8.2 Querying All Elastic Resource Pools

Function

This API is used to query all elastic resource pools.

URI

- URI format

  GET /v3/{project_id}/elastic-resource-pools
- Parameter description

Table 8-5 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 8-6 query parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Page size. The default value is **100**.
name	No	String	Fuzzy match based on the elastic resource pool name.
offset	No	Integer	Offset. The default value is **0**.
status	No	String	Status of the elastic resource pool. Possible values are as follows: • **AVAILABLE** • **SCALING** • **CREATING** • **FAILED**

Parameter	Mandatory	Type	Description
tags	No	String	Query results are filtered by tag.

Request

None

Response

Table 8-7 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the message may be left blank.
count	No	Long	Total number of records.
elastic_resource_pools	No	Array of objects	Elastic resource pool information. For details, see **Table 8-8**.

Table 8-8 elastic_resource_pools parameters

Parameter	Mandatory	Type	Description
owner	No	String	Account used to create an elastic resource pool.
elastic_resource_pool_name	No	String	Elastic resource pool name.
description	No	String	Elastic resource pool description information.
max_cu	No	Integer	Maximum number of CUs.
min_cu	No	Integer	Minimum number of CUs.
actual_cu	No	Integer	Actual number of CUs.

Parameter	Mandatory	Type	Description
cidr_in_vpc	No	String	Subnet information of the elastic resource pool.
create_time	No	Long	Time when the elastic resource pool is created.
update_time	No	Long	Time when the elastic resource pool is updated.
current_cu	No	Integer	Number of current CUs.
status	No	String	Status of the elastic resource pool. Possible values are as follows: • **AVAILABLE** • **SCALING** • **CREATING** • **FAILED**
resource_id	No	String	Resource ID of the elastic resource pool.
fail_reason	No	String	Cause of the elastic resource pool creation failure.
enterprise_project_id	No	String	Enterprise project ID.
prepay_cu	No	Integer	Number of prepaid CUs.
charging_mode	No	Integer	Billing mode.
queues	No	Array of Strings	Information about the queue the elastic resource pool belongs to.
manager	No	String	Type of the elastic resource pool.
label	No	Map<string, string>	Label used to identify the elastic resource pool. Currently, only the developer label is supported. The value is "label": {"billing_spec_code":"developer"}.
id	No	Long	ID of the elastic resource pool

Example Request

None

Example Response

```
{
  "is_success": true,
```



```
 "message": "",
 "count": 2,
 "elastic_resource_pools": [
  {
    "owner": "ei_dlics_d00352221",
    "id": 1026,
    "elastic_resource_pool_name": "elastic_pool_0622_10",
    "description": "",
    "max_cu": 0,
    "min_cu": 0,
    "actual_cu": 0,
    "cidr_in_vpc": "172.16.0.0/12",
    "create_time": 1624366266826,
    "current_cu": 64,
    "status": "AVAILABLE",
    "resource_id": "ecc98d89-3fd1-4dec-b572-aa6e2ef82391",
    "fail_reason": "",
    "enterprise_project_id": "0",
    "prepay_cu": 0,
    "charging_mode": 1
  },
  {
    "owner": "ei_dlics_d00352221",
    "id": 1002,
    "elastic_resource_pool_name": "elastic_pool_0622_0",
    "description": "test",
    "max_cu": 684,
    "min_cu": 0,
    "actual_cu": 0,
    "cidr_in_vpc": "172.16.0.0/12",
    "create_time": 1624353878084,
    "status": "AVAILABLE",
    "resource_id": "ecc98d89-3fd1-4dec-b572-aa6e2ef82391",
    "fail_reason": "",
    "enterprise_project_id": "0",
    "prepay_cu": 0,
    "charging_mode": 1
  }
 ]
}
```


Status Codes

Status Code	Description
200	OK

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

8.3 Deleting an Elastic Resource Pool

Function

This API is used to delete an elastic resource pool.

URI

- URI format

  DELETE /v3/{project_id}/elastic-resource-pools/{elastic_resource_pool_name}
- Parameter description

  Table 8-9 URI parameters

Parameter	Mandatory	Type	Description
elastic_resource_pool_name	Yes	String	Elastic resource pool name.
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

None

Response

Table 8-10 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the message may be left blank.

Example Request

None

Example Response

```
{
  "is_success" : true,
  "message" : ""
}
```


Status Codes

Status Code	Description
200	OK

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

8.4 Modifying Elastic Resource Pool Information

Function

This API is used to modify elastic resource pool information.

URI

- URI format

  PUT /v3/{project_id}/elastic-resource-pools/{elastic_resource_pool_name}
- Parameter description

  Table 8-11 URI parameters

Parameter	Mandatory	Type	Description
elastic_resource_pool_name	Yes	String	Elastic resource pool name.
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 8-12 Request parameters

Parameter	Mandatory	Type	Description
description	No	String	Description. The value can contain a maximum of 256 characters.

Parameter	Mandatory	Type	Description
max_cu	No	Integer	Maximum CUs allowed for an elastic resource pool.
min_cu	No	Integer	Maximum CUs allowed for an elastic resource pool.

Response

Table 8-13 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the message may be left blank.

Example Request

Modify the description, maximum CUs, and minimum CUs of the elastic resource pool. After the modification, the minimum CUs is **78** and the maximum CUs is **990**.

```
{
  "description" : "test_update",
  "min_cu" : 78,
  "max_cu" : 990
}
```

Example Response

```
{
  "is_success" : true,
  "message" : ""
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

8.5 Querying All Queues in an Elastic Resource Pool

Function

This API is used to query all queues in an elastic resource pool.

URI

- URI format

  GET /v3/{project_id}/elastic-resource-pools/{elastic_resource_pool_name}/queues

- Parameter description

Table 8-14 URI parameters

Parameter	Mandatory	Type	Description
elastic_resource_pool_name	Yes	String	Elastic resource pool name.
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 8-15 query parameters

Parameter	Mandatory	Type	Description
limit	No	Integer	Page size. The default value is **100**.
offset	No	Integer	Offset. The default value is **0**.
queue_name	No	String	You can filter data by queue name.

Request

None

Response

Table 8-16 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the message may be left blank.
queues	No	Array of objects	Resource weight of all queues in the elastic resource pool. For details about the parameters, see **Table 8-17**.
count	No	Long	Number of queues bound to the elastic resource pool.

Table 8-17 queues

Parameter	Mandatory	Type	Description
queue_name	No	String	Queue name.
enterprise_project_id	No	String	Enterprise project ID.
queue_type	No	String	Queue type.
queue_scaling_policies	No	Array of objects	Scaling policy. For details about the parameters, see **priority_infos**.
owner	No	String	Queue owner.
create_time	No	Long	Time when a queue is created.
engine	No	String	Queue engine type.

Table 8-18 queue_scaling_policies

Parameter	Mandatory	Type	Description
priority	No	Integer	The policy priority ranges from 1 to 100. The value **100** indicates the highest priority.
impact_start_time	No	String	Time when a policy takes effect.
impact_stop_time	No	String	Time when a policy expires.
min_cu	No	Integer	Minimum number of CUs.
max_cu	No	Integer	Maximum number of CUs.

Example Request

None

Example Response

```
{
  "is_success": true,
  "message": "",
  "count": 1,
  "queues": [
    {
      "queue_name": "lhm_sql",
      "enterprise_project_id": "0",
      "queue_type": "sql",
      "queue_scaling_policies": [
        {
          "priority": 50,
          "impact_start_time": "00:00",
          "impact_stop_time": "24:00",
          "min_cu": 16,
          "max_cu": 16
        },
        {
          "priority": 100,
          "impact_start_time": "00:00",
          "impact_stop_time": "12:00",
          "min_cu": 32,
          "max_cu": 64
        },
        {
          "priority": 50,
          "impact_start_time": "12:00",
          "impact_stop_time": "24:00",
          "min_cu": 16,
          "max_cu": 32
        }
      ]
    }
  ]
}
```


Status Codes

Status Code	Description
200	OK

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

8.6 Associating a Queue with an Elastic Resource Pool

Function

This API is used to associate a queue with an elastic resource pool.

URI

- URI format

  POST /v3/{project_id}/elastic-resource-pools/{elastic_resource_pool_name}/queues
- Parameter description

  Table 8-19 URI parameters

Parameter	Mandatory	Type	Description
elastic_resource_pool_name	Yes	String	Elastic resource pool name.
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 8-20 Request parameter

Parameter	Mandatory	Type	Description
queue_name	Yes	String	Queue name.

Response

Table 8-21 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the message may be left blank.

Example Request

Associate the **lhm_sql** queue with the elastic resource pool.

```
{
  "queue_name" : "lhm_sql"
}
```

Example Response

```
{
  "is_success" : true,
  "message" : ""
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

8.7 Viewing Scaling History of an Elastic Resource Pool

Function

This API is used to view scaling history of an elastic resource pool.

URI

GET /v3/{project_id}/elastic-resource-pools/{elastic_resource_pool_name}/scale-records

Table 8-22 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
elastic_resource_pool_name	Yes	String	Elastic resource pool name The value can contain 1 to 128 characters.

Table 8-23 query parameters

Parameter	Mandatory	Type	Description
start_time	No	Long	Start time of the historical scaling records you want to query. The time must be 30 days earlier than the current time and earlier than the **end_time**. The value is a UNIX timestamp in milliseconds. • If **start_time** is left empty, data generated in the recent seven days before **end_time** will be queried. The **end_time** cannot be later than 30 days after the current time. • If both **start_time** and **end_time** are left empty, data generated in the recent 15 days before the current time will be queried.

Parameter	Mandatory	Type	Description
end_time	No	Long	End time of the historical scaling records. The value cannot be earlier than the **start_time** or later than the current time. The value is a UNIX timestamp in milliseconds. • If **end_time** is left empty, data generated since the **start_time** will be queried. • If both **start_time** and **end_time** are left empty, data generated in the recent 15 days before the current time will be queried.
status	No	String	Scaling status Enumerated values: • **success** • **fail**
offset	No	Integer	Offset Minimum value: **0**
limit	No	Integer	Number of records displayed on a page Minimum value: **0** Maximum value: **100**

Request

None

Response

Table 8-24 Response body parameters

Parameter	Type	Description
count	Integer	Number of elements in the array
items	Array of arrays	Data returned in the array For details, see **Table 8-25**.

Table 8-25 items parameters

Parameter	Type	Description
max_cu	Integer	Maximum number of CUs
min_cu	Integer	Minimum number of CUs
current_cu	Integer	Scaled number of CUs
origin_cu	Integer	Original number of CUs
target_cu	Integer	Target number of CUs
record_time	Long	Operation completion time
status	String	Scaling status, which can be success or failure
fail_reason	String	Failure cause

Example Request

```
GET https://{endpoint}/v3/{project_id}/elastic-resource-pools/{elastic_resource_pool_name}/scale-records?start_time=1650784624000&end_time=1652625304002&status=&limit=20&offset=1
```

Example Response

The following is an example for a successful query:

```
{
  "count" : 1,
  "items" : [ {
    "max_cu" : 64,
    "min_cu" : 16,
    "current_cu" : 16,
    "target_cu" : 16,
    "origin_cu" : 16,
    "record_time" : 1650784624000,
    "status" : "fail",
    "fail_reason" : "Internal error, please contact technical support."
  } ]
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

8.8 Modifying the Scaling Policy of a Queue Associated with an Elastic Resource Pool

Function

This API is used to modify the scaling policy of a queue associated with an elastic resource pool.

URI

- URI format

  PUT /v3/{project_id}/elastic-resource-pools/{elastic_resource_pool_name}/queues/{queue_name}

- Parameter description

  Table 8-26 URI parameters

Parameter	Mandatory	Type	Description
elastic_resource_pool_name	Yes	String	Elastic resource pool name.
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
queue_name	Yes	String	Name of a bound queue.

Request

Table 8-27 Request parameter

Parameter	Mandatory	Type	Description
queue_scaling_policies	Yes	Array of objects	Scaling policy of a queue in an elastic resource pool. A policy contains the period, priority, and CU range. There must be a default scaling policy (period [00:00, 24:00]) for each queue. For details about the parameters, see **Table 8-28**.

Table 8-28 queue_scaling_policies

Parameter	Mandatory	Type	Description
impact_start_time	Yes	String	Time when a policy takes effect.
impact_stop_time	Yes	String	Time when a policy expires.
priority	Yes	Integer	Priority.
min_cu	Yes	Integer	Minimum number of CUs.
max_cu	Yes	Integer	Maximum number of CUs.

Response

Table 8-29 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the message may be left blank.

Example Request

Modify the scaling policy of a queue associated with an elastic resource pool.

```
{
  "queue_scaling_policies" : [ {
    "priority" : 100,
    "impact_start_time" : "10:00",
    "impact_stop_time" : "22:00",
    "min_cu":"64",
    "max_cu":"752"
  }, {
    "priority" : 50,
    "impact_start_time" : "22:00",
    "impact_stop_time" : "10:00",
    "min_cu":"64",
    "max_cu":"752"
  } ]
}
```

Example Response

```
{
  "is_success" : true,
```



```
 "message" : ""
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

9 Queue-related APIs (Recommended)

9.1 Creating a Queue

Function

This API is used to create a queue. The queue will be bound to specified compute resources.

NOTE

It takes 5 to 15 minutes to start a job using a new queue for the first time.

URI

- URI format

  POST /v1.0/{project_id}/queues
- Parameter description

Table 9-1 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 9-2 Request parameters

Parameter	Mandatory	Type	Description
queue_name	Yes	String	Name of a newly created resource queue. The name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_). Length range: 1 to 128 characters. **NOTE** The queue name is case-insensitive. The uppercase letters will be automatically converted to lowercase letters.
queue_type	No	String	Queue type. The options are as follows: ● **sql**: Queues used to run SQL jobs ● **general**: Queues used to run Flink and Spark Jar jobs. **NOTE** If the type is not specified, the default value **sql** is used.
description	No	String	Description of a queue.
cu_count	Yes	Integer	Minimum number of CUs that are bound to a queue. Currently, the value can only be **16**, **64**, or **256**.
charging_mode	No	Integer	Billing mode of a queue. This value can only be set to **1**, indicating that the billing is based on the CUH used.
enterprise_project_id	No	String	Enterprise project ID. The value **0** indicates the default enterprise project. **NOTE** Users who have enabled Enterprise Management can set this parameter to bind a specified project.
platform	No	String	CPU architecture of compute resources. ● x86_64
resource_mode	No	Integer	Queue resource mode. The options are as follows: **0**: shared resource mode **1**: dedicated resource mode

Parameter	Mandatory	Type	Description
labels	No	Array of Strings	Tag information of the queue to be created. Currently, the tag information includes whether the queue is cross-AZ (JSON string). The value can only be **2**, that is, a dual-AZ queue whose compute resources are distributed in two AZs is created.
feature	No	String	Type of image used by the queue. The options are: • **basic**: basic type • **ai**: AI-enhanced (Only the SQL x86_64 dedicated queue supports this option.) The default value is **basic**. **NOTE** AI-enhanced means that the queue is loaded with an AI image that integrates algorithm packages related to AI on top of the basic image.
tags	No	Array of objects	Queue tags for identifying cloud resources. A tag consists of a key and tag value. For details, see **Table 9-3**.

Table 9-3 tags parameters

Parameter	Mandatory	Type	Description
Key	Yes	String	Tag key. **NOTE** A tag key can contain a maximum of 128 characters. Only letters, digits, spaces, and special characters (_.:=+-@) are allowed, but the value cannot start or end with a space or start with **_sys_**.
value	Yes	String	Tag value. **NOTE** A tag value can contain a maximum of 255 characters. Only letters, digits, spaces, and special characters (_.:=+-@) are allowed. The value cannot start or end with a space.

Response

Table 9-4 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
queue_name	No	String	Name of the created queue. **NOTE** The queue name is case-insensitive. The uppercase letters will be automatically converted to lowercase letters.

Example Request

Create a dedicated general-purpose queue named **queue1**, with specifications of 16 CUs and compute resources distributed in two AZs.

```
{
    "queue_name": "queue1",
    "description": "test",
    "cu_count": 16,
    "resource_mode": 1,
    "enterprise_project_id": "0",
    "queue_type": "general",
    "labels": ["multi_az=2"]
}
```

Creating a queue in a specified elastic resource pool

```
{
    "queue_name": "queue2",
    "description": "test_esp",
    "cu_count": 16,
    "resource_mode": 1,
    "enterprise_project_id": "0",
    "queue_type": "general",
    "labels": ["multi_az=2"],
    "elastic_resource_pool_name": "elastic_pool_0622_10"
}
```

Example Response

```
{
  "is_success": true,
  "message": "",
  "queue_name": "queue1"
}
```

Status Codes

Table 9-5 describes the status code.

Table 9-5 Status codes

Status Code	Description
200	The job is created successfully.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

9.2 Deleting a Queue

Function

This API is used to delete a specified queue.

NOTE

If a task is being executed in a specified queue, the queue cannot be deleted.

URI

- URI format

  DELETE /v1.0/{project_id}/queues/{queue_name}
- Parameter description

Table 9-6 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
queue_name	Yes	String	Name of a queue to be deleted.

Request

None

Response

Table 9-7 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Example Request

None

Example Response

```
{
  "is_success": true,
  "message": ""
}
```

Status Codes

Table 9-8 describes the status code.

Table 9-8 Status codes

Status Code	Description
200	Deletion succeeded.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

9.3 Querying All Queues

Function

This API is used to list all queues under the project.

URI

- URI format

  GET/v1.0/{project_id}/queues
- Parameter description

Table 9-9 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 9-10 query parameter description

Parameter	Mandatory	Type	Description
queue_type	No	String	Type of the queue. The options are as follows: • **sql**: queues for SQL jobs • **general**: queues for Flink and Spark Jar jobs • **all**: queues of all types If this parameter is not specified, the default value **sql** is used.
with-priv	No	Boolean	Whether to return permission information.
with-charge-info	No	Boolean	Whether to return billing information.
page-size	No	Integer	Maximum number of lines displayed on each page. The default value is **Integer.MAX_VALUE**, indicating that all results are displayed on one page.
current-page	No	Integer	Current page number. The default value is **1**.

Parameter	Mandatory	Type	Description
order	No	String	Filed based on which queues are ordered The default value is **queue_name_asc** (alphabetically ascending order on queue names). Other options are **queue_name_desc** (alphabetically descending order on queue names), **cu_asc** (ascending order on CUs), and **cu_desc** (descending order on CUs).
tags	No	String	Query results are filtered by tag.

Request

None

Response

Table 9-11 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
queues	No	Array of Object	Queue information For details, see **Table 9-12**.

Table 9-12 queues parameters description

Parameter	Mandatory	Type	Description
queue_name	No	String	Name of a queue.
description	No	String	Queue description.
owner	No	String	User who creates a queue.
create_time	No	Long	Time when the queue is created. The timestamp is expressed in milliseconds.

Parameter	Mandatory	Type	Description
queue_type	No	String	Queue type. • sql • general • all If this parameter is not specified, the default value **sql** is used.
cu_count	No	Integer	Number of compute units (CUs) bound to a queue, that is, the number of CUs in the current queue.
charging_mode	No	Integer	Billing mode of a queue. • Value **0** indicates the default queue. You will be charged based on the amount of data scanned. • Value **1** indicates that queues are billed based on the used CUH.
resource_id	No	String	Resource ID of a queue.
enterprise_project_id	No	String	Enterprise project ID. **0** indicates the default enterprise project. **NOTE** Users who have enabled Enterprise Management can set this parameter to bind a specified project.
cidr_in_vpc	No	String	The VPC CIDR block of the queue. For example, 10.0.0.0/8-24, 172.16.0.0/12-24, and 192.168.0.0/16-24.
cidr_in_mgntsubnet	No	String	CIDR block of the management subnet
cidr_in_subnet	No	String	Subnet CIDR block
resource_mode	No	Integer	Resource mode • 0: Shared queue • 1: Dedicated queue
platform	No	String	CPU architecture of queue compute resources. • x86_64
is_restarting	No	Boolean	Whether to restart the queue. The default value is **false**.

Parameter	Mandatory	Type	Description
labels	No	String	Tag information of the queue to be created, including the JSON string indicating whether the queue is Dual-AZ. Currently, only the value **2** is supported, indicating that two queues are created.
feature	No	String	Type of image used by the queue. The options are as follows: ● **basic**: basic type ● **ai**: AI-enhanced (Only the SQL x86_64 dedicated queue supports this option.) The default value is **basic**. **NOTE** AI-enhanced means that the queue is loaded with an AI image that integrates algorithm packages related to AI on top of the basic image.
resource_type	No	String	Type of the resource to which a queue belongs. ● vm: ECF cluster ● container: container cluster (Kubernetes)
cu_spec	No	Integer	Specifications of a queue. For a yearly/monthly queue, this parameter indicates the CU value of the yearly/monthly part. For a pay-per-use queue, this parameter indicates the initial value when a user purchases a queue.
cu_scale_out_limit	No	Integer	Upper limit of the CU value for elastic scaling of the current queue.
cu_scale_in_limit	No	Integer	Lower limit of the CU value for elastic scaling of the current queue.

Example Request

None

Example Response

```
{
    "is_success": "true",
    "message": "",
    "queues": [
        {
            "queue_name": "test",
            "owner": "testuser",
            "description": "",
```



```
            "create_time": 1562221422671,
            "queue_type": "spark",
            "cu_count": 16,
            "charging_mode": 2,
            "resource_id": "26afb850-d3c9-42c1-81c0-583d1163e80f",
            "enterprise_project_id": "0",
            "cidr_in_vpc": "10.0.0.0/8",
            "cidr_in_subnet": "10.0.0.0/24",
            "cidr_in_mgntsubnet": "10.23.128.0/24",
            "resource_mode": 1,
            "platform": "x86_64",
            "is_restarting": "false",
            "labels": "multi_az=2",
            "resource_type": "vm",
            "cu_spec": 16
        }
    ]
}
```


Status Codes

Table 9-13 describes the status code.

Table 9-13 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

9.4 Viewing Details of a Queue

Function

This API is used to list details of a specific queue in a project.

URI

- URI format

  GET /v1.0/{project_id}/queues/{queue_name}
- Parameter description

Table 9-14 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
queue_name	Yes	String	Specifies the name of a queue to be queried. **NOTE** The queue name is case-insensitive. The uppercase letters will be automatically converted to lowercase letters.

Request

None

Response

Table 9-15 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
queueName	No	String	Name of a queue. **NOTE** The queue name is case-insensitive. The uppercase letters will be automatically converted to lowercase letters.
description	No	String	Queue description.
owner	No	String	User who creates a queue.
create_time	No	Long	Time when the queue is created. The timestamp is expressed in milliseconds.

Parameter	Mandatory	Type	Description
queueType	No	String	Indicates the queue type. • sql • general • all If this parameter is not specified, the default value **sql** is used.
cuCount	No	Integer	Number of compute units (CUs) bound to a queue, that is, the number of CUs in the current queue.
chargingMode	No	integer	Billing mode of a queue. • Value **0** indicates the default queue. You will be charged based on the amount of data scanned. • Value **1** indicates that queues are billed based on the used CUH.
resource_id	No	String	Resource ID of a queue.
resource_mode	No	Integer	Resource mode • **0**: Shared queue • **1**: Dedicated queue
enterprise_project_id	No	String	Enterprise project ID. **0** indicates the default enterprise project. **NOTE** Users who have enabled Enterprise Management can set this parameter to bind a specified project.
resource_type	No	String	Resource type. • vm: ECF cluster • container: container cluster (Kubernetes)
cu_spec	No	Integer	Specifications of a queue. For a yearly/monthly queue, this parameter indicates the CU value of the yearly/monthly part. For a pay-per-use queue, this parameter indicates the initial value when a user purchases a queue.
cu_scale_out_limit	No	Integer	Upper limit of the CU value for elastic scaling of the current queue.
cu_scale_in_limit	No	Integer	Lower limit of the CU value for elastic scaling of the current queue.

Example Request

None

Example Response

```
{
   "is_success": true,
   "message": "",
   "owner": "testuser",
   "description": "",
   "queueName": "test",
   "create_time": 1587613028851,
   "queueType": "general",
   "cuCount": 16,
   "chargingMode": 1,
   "resource_id": "03d51b88-db63-4611-b779-9a72ba0cf58b",
   "resource_mode": 0,
   "enterprise_project_id": "0",
   "resource_type": "vm",
    "cu_spec": 16
}
```

Status Codes

Table 9-16 describes the status code.

Table 9-16 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

9.5 Restarting, Scaling Out, and Scaling In Queues

Function

This API is used to restart, scale out, and scale in queues.

NOTE

Only SQL queues in the **Available** status can be restarted. (The queue status is **Available** only after the SQL job is successfully executed.)

URI

- URI format

  PUT /v1.0/{project_id}/queues/{queue_name}/action
- Parameter description

  Table 9-17 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
queue_name	Yes	String	Name of a queue.

Request

Table 9-18 Request parameters

Parameter	Mandatory	Type	Description
action	Yes	String	Operations to be performed: • **restart**: Restart a service. Only queues for SQL jobs can be restarted. • **scale_out**: Scale out the queue • **scale_in**: Scale in the queue **NOTE** Currently, only **restart**, **scale_out**, and **scale_in** operations are supported.
force	No	Boolean	Specifies whether to forcibly restart the queue. This parameter is optional when **action** is set to **restart**. The default value is **false**.
cu_count	No	Integer	Number of CUs to be scaled in or out. This parameter is optional when **action** is set to **scale_out** or **scale_in**. The value of **cu_count** must be a multiple of 16.

Response

Table 9-19 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_id	No	String	Specifies the job ID returned when **force** is set to **true**.
queue_name	No	String	Name of the queue to be scaled in or out.
result	No	Boolean	Indicates the scaling result.

Example Request

- Restart a queue.

```
{
    "action": "restart",
    "force": "false"
}
```

- Increase the number of CUs of the queue to 16.

```
{
    "action": "scale_out",
    "cu_count": 16
}
```

Example Response

- Set **force** to **false**.

```
{
    "is_success": true,
    "message": "Restart success"
}
```

- Set **force** to **true**.

```
{
    "is_success": true,
    "message": "Submit restart job success, it need some time to cancel jobs, please wait for a while and check job status",
    "job_id": "d90396c7-3a25-4944-ad1e-99c764d902e7"
}
```

- Scaling

```
{
    "queue_name": "myQueue",
    "result": true
}
```


Status Codes

Table 9-20 describes the status code.

Table 9-20 Status codes

Status Code	Description
200	The operation is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 9-21 Error codes

Error Code	Error Message
DLI.0015	Token info for token is null, return.
DLI.0013	X-Auth-Token is not defined in request. It is mandatory. Please define and send the request.

9.6 Creating an Address Connectivity Test Request

Function

This API is used to send an address connectivity test request to a specified queue and insert the test address into the table.

URI

- URI format

  POST /v1.0/{project_id}/queues/{queue_name}/connection-test
- Parameter description

Table 9-22 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
queue_name	Yes	String	Name of a queue.

Request

Table 9-23 Request parameters

Parameter	Mandatory	Type	Description
address	Yes	String	Test address. The format is *IP address or domain name*:*port*.

Response

Table 9-24 Response parameters

Parameter	Mandatory	Type	Description
is_success	Yes	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	Yes	String	System prompt. If execution succeeds, the parameter setting may be left blank.
task_id	Yes	Integer	Request ID

Example Request

Test the connectivity between the queue and the address **iam.*xxx*.com:443**.

```
{
    "address": "iam.xxx.com:443"
}
```

Example Response

```
{
    "is_success": true,
```



```
    "message": "check connectivity to address:iam.xxx.com with port: 443 successfully",
    "task_id": 9
}
```

Status Codes

Table 9-25 describes status codes.

Table 9-25 Status codes

Status Code	Description
200	The job is created successfully.
400	Request failure.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

9.7 Querying Connectivity Test Details of a Specified Address

Function

This API is used to query the connectivity test result after the test is submitted.

URI

- URI format

  GET /v1.0/{project_id}/queues/{queue_name}/connection-test/{task_id}
- Parameter description

  Table 9-26 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
queue_name	Yes	String	Name of a queue.

Parameter	Mandatory	Type	Description
task_id	Yes	String	Job ID. You can call **Creating an Address Connectivity Test Request** to obtain the value.

Request

None

Response

Table 9-27 Response parameters

Parameter	Mandatory	Type	Description
is_success	Yes	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	Yes	String	System prompt. If execution succeeds, the parameter setting may be left blank.
connectivity	Yes	String	Indicates the connectivity test result.

Example Request

None

Example Response

```
{
    "is_success": true,
    "message": "Get node connectivity status successfully for addressId:9",
    "connectivity": "REACHABLE"
}
```

Status Codes

Table 9-28 describes status codes.

Table 9-28 Status codes

Status Code	Description
200	The query is successful.
400	Request failure.

Status Code	Description
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

10 SQL Job-related APIs

10.1 Submitting a SQL Job (Recommended)

Function

This API is used to submit jobs to a queue using SQL statements.

The job types support DDL, DCL, IMPORT, QUERY, and INSERT. The IMPORT function is the same as that described in **Importing Data (Discarded)**. The difference lies in the implementation method.

Additionally, you can use other APIs to query and manage jobs. For details, see the following sections:

- **Querying Job Status**
- **Querying Job Details**
- **Exporting Query Results**
- **Querying All Jobs**
- **Canceling a Job (Recommended)**

NOTE

This API is synchronous if **job_type** in the response message is **DCL**.

URI

- URI format

  POST /v1.0/{project_id}/jobs/submit-job
- Parameter description

Table 10-1 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 10-2 Request parameters

Parameter	Mandatory	Type	Description
sql	Yes	String	SQL statement that you want to execute.
currentdb	No	String	Database where the SQL statement is executed. This parameter does not need to be configured during database creation.
queue_name	No	String	Name of the queue to which a job to be submitted belongs. The name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_).
conf	No	Array of Strings	You can set the configuration parameters for the SQL job in the form of **Key/Value**. For details about the supported configuration items, see **Table 10-3**.
tags	No	Array of objects	Label of a job. For details, see **Table 10-4**.
engine_type	No	String	The default value is **spark**.

Table 10-3 Configuration parameters description

Parameter	Default Value	Description
spark.sql.files.maxRecordsPerFile	0	Maximum number of records to be written into a single file. If the value is zero or negative, there is no limit.

Parameter	Default Value	Description
spark.sql.autoBroadcastJoinThreshold	209715200	Maximum size of the table that displays all working nodes when a connection is executed. You can set this parameter to **-1** to disable the display. **NOTE** Currently, only the configuration unit metastore table that runs the **ANALYZE TABLE COMPUTE statistics noscan** command and the file-based data source table that directly calculates statistics based on data files are supported.
spark.sql.shuffle.partitions	200	Default number of partitions used to filter data for join or aggregation.
spark.sql.dynamicPartitionOverwrite.enabled	false	Whether DLI overwrites the partitions where data will be written into during runtime. If you set this parameter to **false**, all partitions that meet the specified condition will be deleted before data overwrite starts. For example, if you set **false** and use INSERT OVERWRITE to write partition 2021-02 to a partitioned table that has the 2021-01 partition, this partition will be deleted. If you set this parameter to **true**, DLI does not delete partitions before overwrite starts.
spark.sql.files.maxPartitionBytes	134217728	Maximum number of bytes to be packed into a single partition when a file is read.
spark.sql.badRecordsPath	-	Path of bad records.

Parameter	Default Value	Description
spark.sql.legacy.correlated.scalar.query.enabled	false	• **If set to true:** – When there is no duplicate data in a subquery, executing a correlated subquery does not require deduplication from the subquery's result. – If there is duplicate data in a subquery, executing a correlated subquery will result in an error. To resolve this, the subquery's result must be deduplicated using functions such as **max()** or **min()**. • **If set to false:** Regardless of whether there is duplicate data in a subquery, executing a correlated subquery requires deduplicating the subquery's result using functions such as **max()** or **min()**. Otherwise, an error will occur.

Table 10-4 tags parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key **NOTE** A tag key can contain a maximum of 128 characters. Only letters, digits, spaces, and special characters (_.:=+-@) are allowed, but the value cannot start or end with a space or start with **_sys_**.
value	Yes	String	**NOTE** A tag value can contain a maximum of 255 characters. Only letters, digits, spaces, and special characters (_.:=+-@) are allowed. The value cannot start or end with a space.

Response

Table 10-5 Response parameters

Parameter	Mandatory	Type	Description
is_success	Yes	Boolean	Indicates whether the request is successfully sent. Value **true** indicates that the request is successfully sent.
message	Yes	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_id	Yes	String	ID of a job returned after a job is generated and submitted by using SQL statements. The job ID can be used to query the job status and results.
job_type	Yes	String	Job type. The options include: • **DDL** • **DCL** • **IMPORT** • **EXPORT** • **QUERY** • **INSERT**
schema	No	Array of Map	If the statement type is DDL, the column name and type of DDL are displayed.
rows	No	Array of objects	When the statement type is DDL and **dli.sql.sqlasync.enabled** is set to **false**, the execution results are returned directly. However, only a maximum of 1,000 rows can be returned. If there are more than 1,000 rows, obtain the results asynchronously. That is, when submitting the job, set *xxxx* to **true**, and then obtain the results from the job bucket configured by DLI. The path of the results on the job bucket can be obtained from the **result_path** in the return value of the ShowSqlJobStatus API. The full data of the results will be automatically exported to the job bucket.
job_mode	No	String	Job execution mode. The options are as follows: • **async**: asynchronous • **sync**: synchronous

Example Request

Submit a SQL job. The job execution database and queue are **db1** and **default**, respectively. Then, add the tags **workspace=space1** and **jobName=name1** for the job.

```
{
   "currentdb": "db1",
   "sql": "desc table1",
   "queue_name": "default",
   "conf": [
      "dli.sql.shuffle.partitions = 200"
   ],
   "tags": [
         {
           "key": "workspace",
           "value": "space1"
         },
         {
           "key": "jobName",
           "value": "name1"
         }
    ]
}
```

Example Response

```
{
  "is_success": true,
  "message": "",
  "job_id": "8ecb0777-9c70-4529-9935-29ea0946039c",
  "job_type": "DDL",
  "job_mode":"sync",
  "schema": [
    {
      "col_name": "string"
    },
    {
      "data_type": "string"
    },
    {
      "comment": "string"
    }
  ],
  "rows": [
    [
      "c1",
      "int",
      null
    ],
    [
      "c2",
      "string",
      null
    ]
  ]
}
```

Status Codes

Table 10-6 describes the status code.

Table 10-6 Status codes

Status Code	Description
200	Submitted successfully.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

10.2 Canceling a Job (Recommended)

Function

This API is used to cancel a submitted job. If execution of a job completes or fails, this job cannot be canceled.

URI

- URI format

  DELETE /v1.0/{project_id}/jobs/{job_id}
- Parameter description

  Table 10-7 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
job_id	Yes	String	Job ID. You can get the value by calling **Submitting a SQL Job (Recommended)**.

Request

None

Response

Table 10-8 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Example Request

None

Example Response

```
{
  "is_success": true,
  "message": ""
}
```

Status Codes

Table 10-9 describes the status code.

Table 10-9 Status codes

Status Code	Description
200	Canceled.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

10.3 Querying All Jobs

Function

This API is used to query information about all jobs in the current project.

URI

- URI format
  GET /v1.0/{project_id}/jobs
- Parameter description

Table 10-10 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 10-11 query parameter description

Parameter	Mandatory	Type	Description
page-size	No	Integer	Maximum number of jobs displayed on each page. The value range is as follows: [1, 100]. The default value is **50**.
current-page	No	Integer	Current page number. The default value is **1**.
start	No	Long	Queries the jobs executed later than the time specified by this parameter. The time is a UNIX timestamp in milliseconds.
end	No	Long	Queries the jobs executed earlier than the time specified by this parameter. The time is a UNIX timestamp in milliseconds.
job-type	No	String	Type of a job to be queried. Job types include **DDL**, **DCL**, **IMPORT**, **EXPORT**, **QUERY**, **INSERT**, **DATA_MIGRATION**, **UPDATE**, **DELETE**, **RESTART_QUEUE**, and **SCALE_QUEUE**. To query all types of jobs, enter **ALL**.
job-status	No	String	Status of the job to be queried.
job-id	No	String	ID of the job to be queried. You can get the value by calling **Submitting a SQL Job (Recommended)**.
queue_name	No	String	Specifies **queue_name** as the filter to query jobs running on the specified queue.

Parameter	Mandatory	Type	Description
sql_pattern	No	String	Specifies the SQL segment as the filter. It is case insensitive.
order	No	String	Specifies the job sorting mode. The default value is **start_time_desc** (job submission time in descending order). Four sorting modes are supported: **duration_desc** (job running duration in descending order), **duration_asc** (job running duration in ascending order), **start_time_desc** (job submission time in descending order), and **start_time_asc** (job submission time in ascending order).
engine-type	No	String	Engine type.
owner	No	String	User who submits a job.
tags	No	String	Queue tags for the search. You can specify multiple tags in **key=value** format. ● Request with one specified tag For example, GET /v1.0/{project_id}/jobs?tags=k1%3Dv1 In this example, **=** needs to be escaped to **%3D**, **k1** indicates the tag key, and **v1** indicates the tag value. ● Request with more than one tag Separate tags with commas (,) and convert the commas (,) to **%2C**, for example, GET /v1.0/{project_id}/jobs?.tags=k1%3Dv1%2Ck2%3Dv2. The equal sign (=) is escaped to **%3D**. **k1** indicates a tag key, and **v1** indicates the tag value. **k2** indicates another tag key, and **v2** indicates the tag value. Currently, only fuzzy query is supported. Exact query is not supported.

NOTE

The following is an example of the URL containing the **query** parameter:

GET /v1.0/{project_id}/jobs?page-size=*{size}*&current-page=*{page_number}*&start=*{start_time}*&end=*{end_time}*&job-type=*{QUERY}*&queue_name=*{test}*&order=*{duration_desc}*

Request

None

Response

Table 10-12 Response parameters

Parameter	Mandatory	Type	Description
is_success	Yes	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	Yes	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_count	Yes	Integer	Indicates the total number of jobs.
jobs	Yes	Array of objects	Indicates the information about a job. For details, see **Table 10-13**.

Table 10-13 jobs parameters

Parameter	Mandatory	Type	Description
job_id	Yes	String	Job ID.
job_type	Yes	String	Type of a job.
queue_name	Yes	String	Queue to which a job is submitted.
owner	Yes	String	User who submits a job.
start_time	Yes	Long	Time when a job is started. The timestamp is expressed in milliseconds.
duration	Yes	Long	Job running duration (unit: millisecond).
status	Yes	String	Status of a job, including **LAUNCHING**, **RUNNING**, **FINISHED**, **FAILED**, and **CANCELLED**.
input_row_count	No	Long	Number of records scanned during the Insert job execution.
bad_row_count	No	Long	Number of error records scanned during the Insert job execution.
input_size	Yes	Long	Size of scanned files during job execution.

Parameter	Mandatory	Type	Description
result_count	Yes	Integer	Total number of records returned by the current job or total number of records inserted by the Insert job.
database_name	No	String	Name of the database where the target table resides. **database_name** is valid only for jobs of the **Import** and **Export** types.
table_name	No	String	Name of the target table. **table_name** is valid only for jobs of the **Import** and **Export** types.
with_column_header	No	Boolean	Import jobs, which record whether the imported data contains column names.
detail	Yes	String	JSON string of related columns queried using SQL statements.
statement	Yes	String	SQL statements of a job.
message	No	String	System prompt.
end_time	No	Long	Job end time. The timestamp is in milliseconds.
tags	No	Array of objects	Job tags. For details, see **Table 10-14**.
output_byte	No	String	Number of output bytes of a job.
cpu_cost	No	String	Total CPU used by a job.

Table 10-14 tags parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key
value	Yes	String	Tag value

Example Request

None

Example Response

```
{
 "is_success": true,
 "message": "",
```



```
 "job_count": 1,
 "jobs": [
  {

    "detail": "{\"type\":\"struct\",\"fields\":[{\"name\":\"name\",\"type\":\"string\",\"nullable
\":true,\"metadata\":{}},{\"name\":\"age\",\"type\":\"integer\",\"nullable\":true,\"metadata\":{}}]}",
    "duration": 17731,

    "end_time": 1502349821460,
    "input_size": 0,
    "job_id": "37286cc7-0508-4ffd-b636-951c8a5c75de",
    "job_type": "QUERY",
    "message": "",
    "owner": "tenant1",
    "queue_name": "queue1",
    "result_count": 3,
    "start_time": 1502349803729,
    "statement": "select * from t_json_002",
    "status": "FINISHED",
    "with_column_header": false
  }
 ]
}
```


Status Codes

Table 10-15 describes the status code.

Table 10-15 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

10.4 Previewing SQL Job Query Results

Function

This API is used to view the job execution result after a job is executed using SQL query statements. Currently, you can only query execution results of jobs of the **QUERY** type.

This API can be used to view only the first 1000 result records and does not support pagination query. To view all query results, you need to export the query results first. For details, see **Exporting Query Results**.

URI

- URI format

  GET /v1.0/{project_id}/jobs/{job_id}/preview
- Parameter description

Table 10-16 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
job_id	Yes	String	Job ID

Table 10-17 query parameter description

Parameter	Mandatory	Type	Description
page-size	No	Long	Number of result rows. The value ranges from 1 to 1000. The default rate limit is **1000**.
queue-name	No	String	Name of the execution queue for obtaining job results. If this parameter is not specified, the default system queue is used.

NOTE

The following is an example of the URL containing the **query** parameter:

GET /v1.0/{project_id}/jobs/{job_id}/preview?page-size=*{size}*&queue-name=*{queue_name}*

Request

None

Response

Table 10-18 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_id	No	String	Job ID You can get the value by calling **Submitting a SQL Job (Recommended)**.
job_type	No	String	Job type, including **DDL**, **DCL**, **IMPORT**, **EXPORT**, **QUERY**, **INSERT**, **DATA_MIGRATION**, **UPDATE**, **DELETE**, **RESTART_QUEUE** and **SCALE_QUEUE**. Currently, you can only query execution results of jobs of the **QUERY** type.
row_count	No	Integer	Total number of job results.
input_size	No	Long	Amount of data scanned during job execution.
schema	No	Array of Map	Name and type of the job result column.
rows	No	Array of objects	Job results set.

Example Request

None

Example Response

```
{
  "is_success": true,
  "message": "",
  "job_id": "ead0b276-8ed4-4eb5-b520-58f1511e7033",
  "job_type": "QUERY",
  "row_count": 1,
  "input_size": 74,
  "schema": [
    {
      "c1": "int"
    },
    {
      "c2": "string"
    }
  ],
```

```
 "rows": [
   [
     23,
     "sda"
   ]
 ]
}
```

Status Codes

Table 10-19 describes the status code.

Table 10-19 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

10.5 Exporting Query Results

Function

This API is used to export results returned from the query using SQL statements to OBS. Only the query result of **QUERY** jobs can be exported.

- This API is asynchronous.
- Currently, data can be exported only to OBS, and the OBS path must be specified to the folder level. The OBS path cannot contain commas (,). The OBS bucket name cannot end with the regular expression format ".[0-9]+(.*)". Specifically, if the bucket name contains dots (.), the last dot (.) cannot be followed by a digit, for example, "**.12abc" and "**.12".

URI

- URI format

  POST /v1.0/{project_id}/jobs/{job_id}/export-result
- Parameter description

Table 10-20 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
job_id	Yes	String	Job ID.

Request

Table 10-21 Request parameters

Parameter	Mandatory	Type	Description
data_path	Yes	String	Path for storing the exported data. Currently, data can be stored only on OBS. The OBS path cannot contain folders, for example, the **path** folder in the sample request.
compress	No	String	Compression format of exported data. Currently, **gzip**, **bzip2**, and **deflate** are supported. The default value is **none**, indicating that data is not compressed.
data_type	Yes	String	Storage format of exported data. Currently, only CSV and JSON are supported.
queue_name	No	String	Name of the queue that is specified to execute a task. If no queue is specified, the default queue is used.
export_mode	No	String	Export mode. The parameter value can be **ErrorIfExists** or **Overwrite**. If **export_mode** is not specified, this parameter is set to **ErrorIfExists** by default. • **ErrorIfExists**: Ensure that the specified export directory does not exist. If the specified export directory exists, an error is reported and the export operation cannot be performed. • **Overwrite**: If you add new files to a specific directory, existing files will be deleted.

Parameter	Mandatory	Type	Description
with_column_header	No	Boolean	Whether to export column names when exporting CSV and JSON data. ● If this parameter is set to **true**, the column names are exported. ● If this parameter is set to **false**, the column names are not exported. ● If this parameter is left blank, the default value **false** is used.
limit_num	No	Integer	Number of data records to be exported. The default value is **0**, indicating that all data records are exported.
encoding_type	No	String	Encoding format of the data to be exported. The default value is **utf-8**. The options include: ● utf-8 ● gb2312 ● gbk
quote_char	No	String	User-defined quote character. The default value is double quotes ("). This parameter is available and can be set only when **Data Format** is **csv**. Quotation characters are used to identify the beginning and end of text fields when exporting job results, and are used to separate fields. Only one character can be set. This is mainly used to handle data that contains spaces, special characters, or characters that are the same as the delimiter.

Parameter	Mandatory	Type	Description
escape_char	No	String	User-defined escape character. The default value is a backslash (\). This parameter is available and can be set only when **Data Format** is **csv**. If special characters, such as quotation marks, need to be included in the exported results, they can be represented using escape characters (backslash \). Only one character can be set. Common scenarios for using escape characters are: • If there is a third quotation mark between two quotation marks, add an escape character before the third quotation mark to avoid the field content being split. • If there is already an escape character in the data content, add another escape character before the existing one to avoid the original character being used as an escape character.

Response

Table 10-22 Response parameters

Parameter	Mandatory	Type	Description
is_success	Yes	Boolean	Indicates whether the request is successfully sent. Value **true** indicates that the request is successfully sent.
message	Yes	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_id	No	String	ID of a job returned after a job is generated and submitted by using SQL statements. The job ID can be used to query the job status and results.
job_mode	No	String	Job execution mode. The options are as follows: • **async**: asynchronous • **sync**: synchronous

Example Request

Export query results of SQL statements to OBS and stores the results in JSON format.

```
{
  "data_path": "obs://obs-bucket1/path",
  "data_type": "json",
  "compress": "gzip",
  "with_column_header": "true",
  "queue_name": "queue2",
  "limit_num": 10
}
```

Example Response

```
{
  "is_success": true,
  "message": "",
  "job_id": "37a40ef9-86f5-42e6-b4c6-8febec89cc20",
  "job_mode":"async"
}
```

Status Codes

Table 10-23 describes the status code.

Table 10-23 Status codes

Status Code	Description
200	Export successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

10.6 Querying Job Status

Function

This API is used to query the status of a submitted job.

URI

- URI format
  GET /v1.0/{project_id}/jobs/{job_id}/status
- Parameter description

Table 10-24 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
job_id	Yes	String	Job ID.

Request

None

Response

Table 10-25 Response parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	String	System prompt. If execution succeeds, this parameter is left blank.
job_id	String	Job ID. You can get the value by calling **Submitting a SQL Job (Recommended)**.
job_type	String	Type of a job, Includes **DDL**, **DCL**, **IMPORT**, **EXPORT**, **QUERY**, **INSERT**, **DATA_MIGRATION**, **UPDATE**, **DELETE**, **RESTART_QUEUE** and **SCALE_QUEUE**.
job_mode	String	Job execution mode. The options are as follows: • **async**: asynchronous • **sync**: synchronous
queue_name	String	Name of the queue where the job is submitted.
owner	String	User who submits a job.
start_time	Long	Time when a job is started. The timestamp is in milliseconds.
duration	Long	Job running duration (unit: millisecond).
status	String	Status of a job, including **RUNNING**, **SCALING**, **LAUNCHING**, **FINISHED**, **FAILED**, and **CANCELLED**.

Parameter	Type	Description
input_row_count	Long	Number of records scanned during the Insert job execution.
bad_row_count	Long	Number of error records scanned during the Insert job execution.
input_size	Long	Size of scanned files during job execution (unit: byte).
result_count	Integer	Total number of records returned by the current job or total number of records inserted by the Insert job.
database_name	String	Name of the database where the target table resides. **database_name** is valid only for jobs of the **IMPORT EXPORT**, and **QUERY** types.
table_name	String	Name of the target table. **table_name** is valid only for jobs of the **IMPORT EXPORT**, and **QUERY** types.
detail	String	JSON string for information about related columns.
statement	String	SQL statements of a job.
tags	Array of objects	Job tags. For details, see **Table 10-26**.
user_conf	String	JSON string of related columns queried using SQL statements.
result_format	String	Storage format of job results. Currently, only CSV is supported.
result_path	String	OBS path of job results.

Table 10-26 tags parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key
value	Yes	String	Tag value

Example Request

None

Example Response


```
{
  "is_success": true,
  "message": "",
  "job_id": "208b08d4-0dc2-4dd7-8879-ddd4c020d7aa",
  "job_type": "QUERY",
  "job_mode":"async",
  "queue_name": "default",
  "owner": "test",
  "start_time": 1509335108918,
  "duration": 2523,
  "status": "FINISHED",
  "input_size": 22,
  "result_count": 4,
  "database_name":"dbtest",
  "table_name":"tbtest",
  "detail": "{\"type\":\"struct\",\"fields\":[{\"name\":\"id\",\"type\":\"integer\",\"nullable\":true,\"metadata\":{}},{\"name\":\"name\",\"type\":\"string\",\"nullable\":true,\"metadata\":{}}]}",
  "statement": "select * from t1"
}
```


Status Codes

Table 10-27 describes the status code.

Table 10-27 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

10.7 Querying Job Details

Function

This API is used to query details about jobs, including **databasename**, **tablename**, **file size**, and **export mode**.

URI

- URI format

  GET/v1.0/{project_id}/jobs/{job_id}/detail
- Parameter description

Table 10-28 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
job_id	Yes	String	Job ID. You can get the value by calling **Submitting a SQL Job (Recommended)**.

Request

None

Response

Table 10-29 Response parameters

Parameter	Mandatory	Type	Description
is_success	Yes	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	Yes	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_id	Yes	String	Job ID.
owner	Yes	String	User who submits a job.
start_time	Yes	Long	Time when a job is started. The timestamp is in milliseconds.
duration	Yes	Long	Duration for executing the job (unit: millisecond).
export_mode	No	String	Specified export mode during data export and query result saving. Available values are **ErrorIfExists** and **Overwrite**. • **ErrorIfExists**: Ensure that the specified export directory does not exist. If the specified export directory exists, an error is reported and the export operation cannot be performed. • **Overwrite**: If you add new files to a specific directory, existing files will be deleted.

Parameter	Mandatory	Type	Description
data_path	Yes	String	Path to imported or exported files.
data_type	Yes	String	Type of data to be imported or exported. Currently, only CSV and JSON are supported.
database_name	Yes	String	Name of the database where the table, where data is imported or exported, resides.
table_name	Yes	String	Name of the table where data is imported or exported.
with_column_header	No	Boolean	Whether the imported data contains the column name during the execution of an import job.
delimiter	No	String	User-defined data delimiter set when the import job is executed.
quote_char	No	String	User-defined quotation character set when the import job is executed.
escape_char	No	String	User-defined escape character set when the import job is executed.
date_format	No	String	Table date format specified when the import job is executed.
timestamp_format	No	String	Table time format specified when the import job is executed.
compress	No	String	Compression mode specified when the export job is executed.
tags	No	Array of objects	Job tags. For details, see **Table 10-30**.

Table 10-30 tags parameter

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key
value	Yes	String	Tag value

Example Request

None

Example Response

- Querying jobs of the **Import** type

```
{
  "is_success": true,
  "message": "",
  "data_path": "obs://DLI/computeCharging/test.csv",
  "data_type": "json",
  "database_name": "iam_exist",
  "date_format": "yyyy-MM-dd",
  "delimiter": ",",
  "duration": 1623,
  "escape_char": "\\",
  "job_id": "a85d7298-ecef-47f9-bb31-499d2099d112",
  "owner": "iam_exist",
  "quote_char": "\"",
  "start_time": 1517385246111,
  "table_name": "DLI_table20",
  "timestamp_format": "yyyy-MM-dd HH:mm:ss",
  "with_column_header": false
}
```

- Query jobs of the **Export** type

```
{
  "is_success": true,
  "message": "",
  "compress": "none",
  "data_path": "obs://xxx/dli/path6",
  "data_type": "json",
  "database_name": "submitjob",
  "duration": 4142,
  "export_mode": "Overwrite",
  "job_id": "b89fccb2-de6a-4c6c-b9b2-21f08a2eb85e",
  "owner": "test",
  "start_time": 1524107798024,
  "table_name": "autotest"
}
```

Status Codes

Table 10-31 describes the status code.

Table 10-31 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

10.8 Checking SQL Syntax

Function

This API is used to check the SQL syntax.

URI

- URI format
  POST /v1.0/{project_id}/jobs/check-sql
- Parameter description

Table 10-32 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 10-33 Request parameters

Parameter	Mandatory	Type	Description
sql	Yes	String	SQL statement that you want to execute.
currentdb	No	String	Database where the SQL statement is executed. **NOTE** • If the SQL statement contains **db_name**, for example, ***select * from db1.t1***, you do not need to set this parameter. • If the SQL statement does not contain **db_name**, the semantics check will fail when you do not set this parameter or set this parameter to an incorrect value.

Response

Table 10-34 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_type	No	String	Type of a job. Job types include the following: **DDL**, **DCL**, **IMPORT**, **EXPORT**, **QUERY**, and **INSERT**.

Example Request

Check the syntax of the SQL statement **select * from t1**.

```
{
   "currentdb": "db1",
   "sql": "select * from t1"
}
```

Example Response

```
{
  "is_success": true,
  "message": "the sql is ok",
  "job_type":"QUERY"
}
```

Status Codes

Table 10-35 describes the status code.

Table 10-35 Status codes

Status Code	Description
200	The request is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

10.9 Querying the Job Execution Progress

Function

This API is used to obtain the job execution progress. If a job is being executed, information about its subjobs can be obtained. If a job has just started or has ended, information about its subjobs cannot be obtained.

URI

- URI format

  GET /v1/{project_id}/jobs/{job_id}/progress

- Parameter description

  Table 10-36 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
job_id	Yes	String	Job ID

Request

None

Response

Table 10-37 Response parameters

Parameter	Mandatory	Type	Description
is_success	Yes	Boolean	Indicates whether the request is successfully sent. Value **true** indicates that the request is successfully sent.
message	Yes	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_id	No	String	ID of a job returned after a job is generated and submitted by using SQL statements. The job ID can be used to query the job status and results.
status	Yes	String	Job status. The status can be **RUNNING**, **SCALING**, **LAUNCHING**, **FINISHED**, **FAILED**, or **CANCELLED**.

Parameter	Mandatory	Type	Description
sub_job_id	No	Integer	ID of a subjob that is running. If the subjob is not running or it is already finished, the subjob ID may be empty.
progress	No	Double	Progress of a running subjob or the entire job. The value can only be a rough estimate of the subjob progress and does not indicate the detailed job progress. • If the job is just started or being submitted, the progress is displayed as **0**. If the job execution is complete, the progress is displayed as **1**. In this case, progress indicates the running progress of the entire job. Because no subjob is running, **sub_job_id** is not displayed. • If a subjob is running, the running progress of the subjob is displayed. The calculation method of progress is as follows: Number of completed tasks of the subjob/Total number of tasks of the subjob. In this case, progress indicates the running progress of the subjob, and **sub_job_id** indicates the subjob ID.
sub_jobs	No	Array of Object	Details about a subjob of a running job. A job may contain multiple subjobs. For details, see **Table 10-38**.

Table 10-38 Parameters in the **sub_jobs** field

Parameter	Mandatory	Type	Description
id	No	Integer	Subjob ID, corresponding to **jobId** of the open-source spark JobData.
name	No	String	Subjob name, corresponding to the **name** of the open-source spark JobData.
description	No	String	Description of a subjob, corresponding to the **description** of the open-source spark JobData.
submission_time	No	String	Submission time of a subjob, corresponding to the **submissionTime** of open-source Spark JobData.
completion_time	No	String	Completion time of a subjob, corresponding to the **completionTime** of the open-source Spark JobData.

Parameter	Mandatory	Type	Description
stage_ids	No	Array of Integer	Stage ID of the subjob, corresponding to the **stageIds** of the open-source spark JobData.
job_group	No	String	ID of a DLI job, corresponding to the **jobGroup** of open-source Spark JobData.
status	No	String	Subjob status, corresponding to the **status** of open-source spark JobData.
num_tasks	No	Integer	Number of subjobs, corresponding to **numTasks** of the open-source Spark JobData.
num_active_tasks	No	Integer	Number of running tasks in a subjob, corresponding to **numActiveTasks** of the open-source Spark JobData.
num_completed_tasks	No	Integer	Number of tasks that have been completed in a subjob, corresponding to **numCompletedTasks** of open-source Spark JobData.
num_skipped_tasks	No	Integer	Number of tasks skipped in a subjob, corresponding to **numSkippedTasks** of open-source Spark JobData.
num_failed_tasks	No	Integer	Number of subtasks that fail to be skipped, corresponding to **numFailedTasks** of open-source Spark JobData.
num_killed_tasks	No	Integer	Number of tasks killed in the subjob, corresponding to **numKilledTasks** of the open-source Spark JobData.
num_completed_indices	No	Integer	Subjob completion index, corresponding to the **numCompletedIndices** of the open-source Spark JobData.
num_active_stages	No	Integer	Number of stages that are running in the subjob, corresponding to **numActiveStages** of the open-source Spark JobData.
num_completed_stages	No	Integer	Number of stages that have been completed in the subjob, corresponding to **numCompletedStages** of the open-source Spark JobData.
num_skipped_stages	No	Integer	Number of stages skipped in the subjob, corresponding to **numSkippedStages** of the open-source Spark JobData.

Parameter	Mandatory	Type	Description
num_failed_stages	No	Integer	Number of failed stages in a subjob, corresponding to **numFailedStages** of the open-source Spark JobData.
killed_tasks_summary	No	Map<string,integer>	Summary of the killed tasks in the subjob, corresponding to **killedTasksSummary** of open-source spark JobData.

Example Request

None

Example Response


```
{
   "is_success": true,
   "message": "",
   "job_id": "85798b38-ae44-48eb-bb90-7cf0dcdafe7b",
   "status": "RUNNING",
   "sub_job_id": 0,
   "progress": 0,
   "sub_jobs": [
      {
         "id": 0,
         "name": "runJob at FileFormatWriter.scala:266",
         "submission_time": "Mon Jul 27 17:24:03 CST 2020",
         "stage_ids": [
            0
         ],
         "job_group": "85798b38-ae44-48eb-bb90-7cf0dcdafe7b",
         "status": "RUNNING",
         "num_tasks": 1,
         "num_active_tasks": 1,
         "num_completed_tasks": 0,
         "num_skipped_tasks": 0,
         "num_failed_tasks": 0,
         "num_killed_tasks": 0,
         "num_completed_indices": 0,
         "num_active_stages": 1,
         "num_completed_stages": 0,
         "num_skipped_stages": 0,
         "num_failed_stages": 0
      }
   ]
}
```


Status Codes

Table 10-39 describes the status code.

Table 10-39 Status codes

Status Code	Description
200	The query is successful.

Status Code	Description
400	Request error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 10-40 Error codes

Error Code	Error Message
DLI.0999	The queue backend version is too old or the queue is busy.

11 SQL Template-related APIs

11.1 Saving a SQL template

Function

This API is used to store specified SQL statements for future reuse.

URI

- URI format

  POST /v1.0/{project_id}/sqls
- Parameter description

  Table 11-1 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 11-2 Parameters

Parameter	Mandatory	Type	Description
sql	Yes	String	New SQL template.

Parameter	Mandatory	Type	Description
sql_name	Yes	String	Name of the SQL template you want to save. The name must be unique in the current project.
description	No	String	Description of the SQL template. The value can be an empty string.
group	No	String	Name of the template group

Response

Table 11-3 Parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successful
message	String	System prompt. If execution succeeds, the message may be left blank. If the execution fails, the value is the cause of the failure.
sql_id	String	ID of the SQL template
group	String	Name of the template group

Example Request

Create a SQL template.

```
{
    "sql": "select * from t1",
    "sql_name": "sql1",
    "description": "use to select t1",
    "group": ""
}
```

Example Response

```
{
  "is_success": true,
  "message": "",
  "sql_id": "1994b029-3f76-4281-9e35-e370cb4ee26d1501998934432"
}
```


Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

11.2 Checking All SQL Templates

Function

This API is used to check all SQL templates saved by a user.

URI

- URI format

  GET /v1.0/{project_id}/sqls
- Parameter description

Table 11-4 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 11-5 query parameters

Parameter	Mandatory	Type	Description
keyword	No	String	Keyword of SQL template names

Request

None

Response

Table 11-6 Response parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successful.
message	String	System prompt. If execution succeeds, the message may be left blank. If the execution fails, the value will be the cause of the failure.
sql_count	Int	Total number of SQL templates
sqls	Array of Object	SQL template information. For details, see **Table 11-7**.

Table 11-7 sqls parameters

Parameter	Type	Description
sql_id	String	SQL template ID
sql_name	String	Name of an SQL template
sql	String	SQL template text
description	String	Description of an SQL template
owner	String	Creator of the SQL template
group	String	Name of the template group

Example Request

None

Example Response

```
{
  "is_success": true,
  "message": "",
  "sql_count": 1,
  "sqls": [
    {
      "description": "use to select t1",
      "owner": "tenant1",
      "sql": "select * from t1",
      "sql_id": "1994b029-3f76-4281-9e35-e370cb4ee26d1501998934432",
      "sql_name": "sql1",
      "group": ""
    }
  ]
}
```


Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

11.3 Updating a SQL template

Function

This API is used to update an SQL template.

URI

- URI format
  PUT /v1.0/{project_id}/sqls/{sql_id}
- Parameter description

Table 11-8 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
sql_id	Yes	String	SQL template ID

Request

Table 11-9 Request parameters

Parameter	Mandatory	Type	Description
sql	No	String	Updated SQL template text.
sql_name	No	String	Name of the updated SQL template. The name must be unique in the current project.

Parameter	Mandatory	Type	Description
description	No	String	Description of an SQL template. This parameter can be left blank.
group	No	String	SQL template group information.

Response

Table 11-10 Response parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successful.
message	String	System prompt. If execution succeeds, the message may be left blank. If the execution fails, the value will be the cause of the failure.

Example Request

Update information about the SQL template, including the SQL statements, template name, template description, and template group information.

```
{
  "sql": "select * from t1",
  "sql_name": "sql1",
  "description": "use to select t1",
  "group": ""
}
```

Example Response

```
{
  "is_success": true,
"message": ""
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

11.4 Deleting a SQL template

Function

This API is used to delete SQL templates in batches.

URI

- URI format

  POST /v1.0/{project_id}/sqls-deletion
- Parameter description

  Table 11-11 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 11-12 Request parameter

Parameter	Mandatory	Type	Description
sql_ids	Yes	Array<String>	IDs of SQL templates you want to delete.

Response

Table 11-13 Response parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successful.
message	String	System prompt. If execution succeeds, the message may be left blank. If the execution fails, the value will be the cause of the failure.

Example Request

Delete the SQL template whose ID is **27b79c50-4ac6-4050-8b81-8f0dbd7be184**.

```
{
    "sql_ids": ["27b79c50-4ac6-4050-8b81-8f0dbd7be184"]
}
```

Example Response

```
{
  "is_success": true,
  "message": ""
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

12 Flink Job-related APIs

12.1 Creating a SQL Job

Function

This API is used to create a Flink streaming SQL job.

URI

- URI format
  POST /v1.0/{project_id}/streaming/sql-jobs
- Parameter description

Table 12-1 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 12-2 Request parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the job. The value can contain 1 to 57 characters.
desc	No	String	Job description. Length range: 0 to 512 characters.

Parameter	Mandatory	Type	Description
template_id	No	Integer	Template ID. If both **template_id** and **sql_body** are specified, **sql_body** is used. If **template_id** is specified but **sql_body** is not, fill **sql_body** with the **template_id** value.
queue_name	No	String	Name of a queue. The value can contain 0 to 128 characters.
sql_body	No	String	Stream SQL statement, which includes at least the following three parts: source, query, and sink. Length range: 1,024 x 1,024 characters.
run_mode	No	String	Job running mode. The options are as follows: • **shared_cluster**: indicates that the job is running on a shared cluster. • **exclusive_cluster**: indicates that the job is running on an exclusive cluster. • **edge_node**: indicates that the job is running on an edge node. The default value is **shared_cluster**.
cu_number	No	Integer	Number of CUs selected for a job. The default value is **2**. Sum of the number of compute units and job manager CUs of DLI. CU is also the billing unit of DLI. One CU equals one vCPU and 4 GB of memory. The value is the number of CUs required for job running and cannot exceed the number of CUs in the bound queue. For details about how to set the number of CUs of JobManager, see **manager_cu_number**.

Parameter	Mandatory	Type	Description
parallel_number	No	Integer	Number of parallel jobs set by a user. The default value is **1**. Number of Flink SQL jobs that run at the same time. Properly increasing the number of parallel threads improves the overall computing capability of the job. However, the switchover overhead caused by the increase of threads must be considered. This value cannot be greater than four times the compute units (number of CUs minus the number of JobManager CUs). For details about how to set the number of JobManager CUs, see **manager_cu_number**.
checkpoint_enabled	No	Boolean	Whether to enable the automatic job snapshot function. ● **true**: indicates to enable the automatic job snapshot function. ● **false**: indicates to disable the automatic job snapshot function. ● Default value: **false**
checkpoint_mode	No	Integer	Snapshot mode. There are two options: ● **1**: **ExactlyOnce**, indicates that data is processed only once. ● **2**: **AtLeastOnce**, indicates that data is processed at least once. The default value is **1**.
checkpoint_interval	No	Integer	Snapshot interval. The unit is second. The default value is **10**.
obs_bucket	No	String	OBS bucket where users are authorized to save the snapshot. This parameter is valid only when **checkpoint_enabled** is set to **true**. OBS bucket where users are authorized to save the snapshot. This parameter is valid only when **log_enabled** is set to **true**.
log_enabled	No	Boolean	Whether to enable the function of uploading job logs to users' OBS buckets. The default value is **false**.
smn_topic	No	String	SMN topic. If a job fails, the system will send a message to users subscribed to the SMN topic.

Parameter	Mandatory	Type	Description
restart_when_exception	No	Boolean	Whether to enable the function of automatically restarting a job upon job exceptions. The default value is **false**.
idle_state_retention	No	Integer	Retention time of the idle state. The unit is second. The default value is **3600**.
job_type	No	String	Job type. The options include **flink_sql_job** and **flink_opensource_sql_job**. The default value is **flink_opensource_sql_job**. ● If **run_mode** is set to **exclusive_cluster**, **job_type** must be set to **flink_sql_job** or **flink_opensource_sql_job**. ● If **run_mode** is set to **shared_cluster**, **job_type** must be set to **flink_sql_job**.
dirty_data_strategy	No	String	Dirty data policy of a job. ● **2:obsDir**: Save. **obsDir** specifies the path for storing dirty data. ● **1**: Trigger a job exception ● **0**: Ignore The default value is **0**.
udf_jar_url	No	String	Name of the resource package that has been uploaded to the DLI resource management system. The UDF Jar file of the SQL job is specified by this parameter.
manager_cu_number	No	Integer	Number of CUs in the JobManager selected for a job. The default value is **1**.
tm_cus	No	Integer	Number of CUs for each TaskManager. The default value is **1**.
tm_slot_num	No	Integer	Number of slots in each TaskManager. The default value is **(parallel_number*tm_cus)/(cu_number-manager_cu_number)**.
resume_checkpoint	No	Boolean	Whether the abnormal restart is recovered from the checkpoint.
resume_max_num	No	Integer	Maximum number of retry times upon exceptions. The unit is times/hour. Value range: -1 or greater than 0. The default value is **-1**, indicating that the number of times is unlimited.

Parameter	Mandatory	Type	Description
tags	No	Array of objects	Label of a Flink SQL job. For details, see Table 12-3.
runtime_config	No	String	Customizes optimization parameters when a Flink job is running.
flink_version	No	String	Flink version.

Table 12-3 tags parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key **NOTE** A tag key can contain a maximum of 36 characters. Only letters, digits, hyphens (-), underscores (_), and spaces are allowed. The key cannot start or end with a space.
value	Yes	String	Tag value **NOTE** A tag value can contain a maximum of 43 characters. Only letters, digits, hyphens (-), underscores (_), periods (.), and spaces are allowed. The value cannot start or end with a space.

Response

Table 12-4 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	String	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	Message content.
job	No	Object	Information about the job status. For details, see Table 12-5.

Table 12-5 job parameters

Parameter	Mandatory	Type	Description
job_id	Yes	Long	Job ID.
status_name	No	String	Name of job status. For details, see the description of the **status** field in **Querying Job Details**.
status_desc	No	String	Status description. Causes and suggestions for the abnormal status.

Example Request

Use the template whose ID is **100000** to create a Flink SQL job named **myjob**. The job runs in dedicated mode on the **testQueue** queue.

```
{
    "name": "myjob",
    "desc": "This is a job used for counting characters.",
    "template_id": 100000,
    "queue_name": "testQueue",
    "sql_body": "select * from source_table",
    "run_mode": "exclusive_cluster",
    "cu_number": 2,
    "parallel_number": 1,
    "checkpoint_enabled": false,
    "checkpoint_mode": "exactly_once",
    "checkpoint_interval": 0,
    "obs_bucket": "my_obs_bucket",
    "log_enabled": false,
    "restart_when_exception": false,
    "idle_state_retention": 3600,
    "job_type": "flink_sql_job",
    "dirty_data_strategy": "0",
    "udf_jar_url": "group/test.jar"
}
```

Example Response

```
{
    "is_success": "true",
    "message": "A DLI job is created successfully.",
    "job": {
        "job_id": 148,
        "status_name": "job_init",
        "status_desc": ""
    }
}
```

Status Codes

Table 12-6 describes status codes.

Table 12-6 Status codes

Status Code	Description
200	The job is created successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

12.2 Updating a SQL Job

Function

This API is used to modify a Flink SQL job.

URI

- URI format

  PUT /v1.0/{project_id}/streaming/sql-jobs/{job_id}

- Parameter description

Table 12-7 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
job_id	Yes	Long	Job ID. Refer to **Creating a SQL Job** to obtain the value.

Request

Table 12-8 Request parameters

Parameter	Mandatory	Type	Description
name	No	String	Name of a job. Length range: 0 to 57 characters.

Parameter	Mandatory	Type	Description
desc	No	String	Job description. Length range: 0 to 512 characters.
queue_name	No	String	Name of a queue. The value can contain 0 to 128 characters.
sql_body	No	String	Stream SQL statement, which includes at least the following three parts: source, query, and sink. Length range: 0 to 1024x1024 characters.
run_mode	No	String	Job running mode. The options are as follows: ● **shared_cluster**: indicates that the job is running on a shared cluster. ● **exclusive_cluster**: indicates that the job is running on an exclusive cluster. ● **edge_node**: indicates that the job is running on an edge node. The default value is **shared_cluster**.
cu_number	No	Integer	Number of CUs selected for a job. The default value is **2**.
parallel_number	No	Integer	Number of parallel jobs set by a user. The default value is **1**.
checkpoint_enabled	No	Boolean	Whether to enable the automatic job snapshot function. ● **true**: indicates to enable the automatic job snapshot function. ● **false**: indicates to disable the automatic job snapshot function. ● Default value: **false**
checkpoint_mode	No	Integer	Snapshot mode. There are two options: ● **1**: **ExactlyOnce**, indicates that data is processed only once. ● **2**: **at_least_once**, indicates that data is processed at least once. The default value is **1**.

Parameter	Mandatory	Type	Description
checkpoint_interval	No	Integer	Snapshot interval. The unit is second. The default value is **10**.
obs_bucket	No	String	OBS bucket where users are authorized to save the snapshot. This parameter is valid only when **checkpoint_enabled** is set to **true**. OBS bucket where users are authorized to save the snapshot. This parameter is valid only when **log_enabled** is set to **true**.
log_enabled	No	Boolean	Whether to enable the function of uploading job logs to users' OBS buckets. The default value is **false**.
smn_topic	No	String	SMN topic. If a job fails, the system will send a message to users subscribed to the SMN topic.
restart_when_exception	No	Boolean	Whether to enable the function of automatically restarting a job upon job exceptions. The default value is **false**.
idle_state_retention	No	Integer	Expiration time, in seconds. The default value is **3600**.
edge_group_ids	No	Array of Strings	List of edge computing group IDs. Use commas (,) to separate multiple IDs.
dirty_data_strategy	No	String	Dirty data policy of a job. ● **2:obsDir**: Save. **obsDir** specifies the path for storing dirty data. ● **1**: Trigger a job exception ● **0**: Ignore The default value is **0**.
udf_jar_url	No	String	Name of the resource package that has been uploaded to the DLI resource management system. The UDF Jar file of the SQL job is specified by this parameter.
manager_cu_number	No	Integer	Number of CUs in the JobManager selected for a job. The default value is **1**.

Parameter	Mandatory	Type	Description
tm_cus	No	Integer	Number of CUs for each TaskManager. The default value is **1**.
tm_slot_num	No	Integer	Number of slots in each TaskManager. The default value is **(parallel_number*tm_cus)/(cu_number-manager_cu_number)**.
operator_config	No	String	Degree of parallelism (DOP) of an operator.
resume_checkpoint	No	Boolean	Whether the abnormal restart is recovered from the checkpoint.
resume_max_num	No	Integer	Maximum number of retry times upon exceptions. The unit is times/hour. Value range: -1 or greater than 0. The default value is **-1**, indicating that the number of times is unlimited.
static_estimator_config	No	String	Traffic or hit ratio of each operator, which is a string in JSON format. Example: {"operator_list":[{"id":"0a448493b4782967b150582570326227","rate_factor":0.55},{"id":"6d2677a0ecc3fd8df0b72ec675edf8f4","rate_factor":1},{"id":"ea632d67b7d595e5b851708ae9ad79d6","rate_factor":0.55},{"id":"bc764cd8ddf7a0cff126f51c16239658","output_rate":2000}]}
runtime_config	No	String	Customizes optimization parameters when a Flink job is running.
flink_version	No	String	Flink version. Currently, only 1.10 and 1.12 are supported.

Response

Table 12-9 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	String	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	Message content.
job	No	Object	Information about job update. For details, see **Table 12-10**.

Table 12-10 job parameter

Parameter	Mandatory	Type	Description
update_time	No	Long	Job update time, expressed by milliseconds

Example Request

Update an existing SQL job. The updated job is named **myjob** and runs on **testQueue** in shared mode.

```
{
    "name": "myjob",
    "desc": "My first job",
    "queue_name": "testQueue",
    "sql_body": "select * from source_table",
    "run_mode": "shared_cluster",
    "cu_number": 4,
    "parallel_number": 4,
    "checkpoint_enabled": false,
    "checkpoint_mode": "exactly_once",
    "checkpoint_interval": 10,
    "obs_bucket": "",
    "log_enabled": false,
    "smn_topic": "",
    "restart_when_exception": false,
    "idle_state_retention": 3600,
    "edge_group_ids": [
        "62de1e1c-066e-48a8-a79d-f461a31b2ee1",
        "2eb00f85-99f2-4144-bcb7-d39ff47f9002"
    ],
    "dirty_data_strategy": "0",
    "udf_jar_url": "group/test.jar"
}
```

Example Response

```
{
    "is_success": "true",
```



```
    "message": "The job is updated successfully.",
    "job": {
        "update_time": 1578905682534
    }
}
```

Status Codes

Table 12-11 describes status codes.

Table 12-11 Status codes

Status Code	Description
200	The job is updated successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

12.3 Creating a Flink Jar job

Function

This API is used to create custom jobs, which currently support the JAR format and run in dedicated queues.

URI

- URI format

  POST /v1.0/{project_id}/streaming/flink-jobs
- Parameter description

  Table 12-12 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 12-13 Parameter description

Parameter	Mandatory	Type	Description
name	Yes	String	Name of the job. The value can contain 1 to 57 characters.
desc	No	String	Job description. Length range: 0 to 512 characters.
queue_name	No	String	Name of a queue. The value can contain 0 to 128 characters.
cu_number	No	Integer	Number of CUs selected for a job.
manager_cu_number	No	Integer	Number of CUs on the management node selected by the user for a job, which corresponds to the number of Flink job managers. The default value is **1**.
parallel_number	No	Integer	Number of parallel operations selected for a job.
log_enabled	No	Boolean	Whether to enable the job log function. • **true**: indicates to enable the job log function. • **false**: indicates to disable the job log function. • Default value: **false**
obs_bucket	No	String	OBS bucket where users are authorized to save logs when **log_enabled** is set to **true**.
smn_topic	No	String	SMN topic. If a job fails, the system will send a message to users subscribed to the SMN topic.
main_class	No	String	Job entry class.
entrypoint_args	No	String	Job entry parameter. Multiple parameters are separated by spaces.
restart_when_exception	No	Boolean	Whether to enable the function of restart upon exceptions. The default value is **false**.

Parameter	Mandatory	Type	Description
entrypoint	No	String	Name of the package that has been uploaded to the DLI resource management system. This parameter is used to customize the JAR file where the job main class is located.
dependency_jars	No	Array of Strings	Name of the package that has been uploaded to the DLI resource management system. This parameter is used to customize other dependency packages. Example: **myGroup/test.jar,myGroup/test1.jar**.
dependency_files	No	Array of Strings	Name of the resource package that has been uploaded to the DLI resource management system. This parameter is used to customize dependency files. Example: **myGroup/test.cvs,myGroup/test1.csv**. 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. `ClassName.class.getClassLoader().getResource("userData/fileName")`
tm_cus	No	Integer	Number of CUs for each TaskManager. The default value is **1**.
tm_slot_num	No	Integer	Number of slots in each TaskManager. The default value is **(parallel_number*tm_cus)/(cu_number-manager_cu_number)**.
feature	No	String	Job feature. Type of the Flink image used by a job. • **basic**: indicates that the basic Flink image provided by DLI is used. • **custom**: indicates that the user-defined Flink image is used.

Parameter	Mandatory	Type	Description
flink_version	No	String	Flink version. This parameter is valid only when **feature** is set to **basic**. You can use this parameter with the **feature** parameter to specify the version of the DLI basic Flink image used for job running.
image	No	String	Custom image. The format is **Organization name/Image name:Image version**. This parameter is valid only when **feature** is set to **custom**. You can use this parameter with the **feature** parameter to specify a user-defined Flink image for job running. For details about how to use custom images, see .
resume_checkpoint	No	Boolean	Whether the abnormal restart is recovered from the checkpoint.
resume_max_num	No	Integer	Maximum number of retry times upon exceptions. The unit is times/hour. Value range: -1 or greater than 0. The default value is **-1**, indicating that the number of times is unlimited.
checkpoint_path	No	String	Storage address of the checkpoint in the JAR file of the user. The path must be unique.
tags	No	Array of objects	Label of a Flink JAR job. For details, see **Table 12-14**.
runtime_config	No	String	Customizes optimization parameters when a Flink job is running.

Table 12-14 tags parameter

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key. **NOTE** A tag key can contain a maximum of 36 characters. Special characters (=*<>|) are not allowed, and the key cannot start with a space.

Parameter	Mandatory	Type	Description
value	Yes	String	Tag key. **NOTE** A tag value can contain a maximum of 43 characters. Special characters (=*<>|) are not allowed, and the value cannot start with a space.

Response

Table 12-15 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	String	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	Message content.
job	No	Object	Information about the job status. For details, see **Table 12-16**.

Table 12-16 job parameters

Parameter	Mandatory	Type	Description
job_id	Yes	Long	Job ID.
status_name	No	String	Name of job status.
status_desc	No	String	Status description. Causes and suggestions for the abnormal status.

Example Request

Create a Flink Jar job named **test**, set the job to be executed on **testQueue**, set the number of CUs used by the job, and enable the job log function.

```
{
    "name": "test",
    "desc": "job for test",
    "queue_name": "testQueue",
    "manager_cu_number": 1,
    "cu_number": 2,
    "parallel_number": 1,
    "tm_cus": 1,
    "tm_slot_num": 1,
```



```
  "log_enabled": true,
  "obs_bucket": "bucketName",
  "smn_topic": "topic",
  "main_class": "org.apache.flink.examples.streaming.JavaQueueStream",
  "restart_when_exception": false,
  "entrypoint": "javaQueueStream.jar",
  "entrypoint_args":"-windowSize 2000 -rate 3",
  "dependency_jars": [
      "myGroup/test.jar",
      "myGroup/test1.jar"
  ],
  "dependency_files": [
      "myGroup/test.csv",
      "myGroup/test1.csv"
  ]
}
```

Example Response

```
{
  "is_success": true,
  "message": "A Flink job is created successfully.",
  "job": {
    "job_id": 138,
    "status_name": "job_init",
    "status_desc": ""
  }
}
```

Status Codes

Table 12-17 describes status codes.

Table 12-17 Status codes

Status Code	Description
200	The custom Flink job is created successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

12.4 Updating a Flink Jar Job

Function

This API is used to update custom jobs, which currently support the JAR format and run in dedicated queues.

URI

- URI format

  PUT /v1.0/{*project_id*}/streaming/flink-jobs/{*job_id*}
- Parameter description

  Table 12-18 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
job_id	Yes	Long	Job ID. Refer to **Creating a Flink Jar job** to obtain the value.

Request

Table 12-19 Parameter description

Parameter	Mandatory	Type	Description
name	No	String	Name of the job. Length range: 0 to 57 characters.
desc	No	String	Job description. Length range: 0 to 512 characters.
queue_name	No	String	Name of a queue. Length range: 1 to 128 characters.
cu_number	No	Integer	Number of CUs selected for a job. The default value is **2**.
manager_cu_number	No	Integer	Number of CUs on the management node selected by the user for a job, which corresponds to the number of Flink job managers. The default value is **1**.
parallel_number	No	Integer	Number of parallel operations selected for a job. The default value is **1**.
log_enabled	No	Boolean	Whether to enable the job log function. • **true**: indicates to enable the job log function. • **false**: indicates to disable the job log function. • Default value: **false**

Parameter	Mandatory	Type	Description
obs_bucket	No	String	OBS bucket where users are authorized to save logs when **log_enabled** is set to **true**.
smn_topic	No	String	SMN topic. If a job fails, the system will send a message to users subscribed to the SMN topic.
main_class	No	String	Job entry class.
entrypoint_args	No	String	Job entry parameter. Multiple parameters are separated by spaces.
restart_when_exception	No	Boolean	Whether to enable the function of restart upon exceptions. The default value is **false**.
entrypoint	No	String	Name of the package that has been uploaded to the DLI resource management system. This parameter is used to customize the JAR file where the job main class is located.
dependency_jars	No	Array of Strings	Name of the package that has been uploaded to the DLI resource management system. This parameter is used to customize other dependency packages. Example: **myGroup/test.jar,myGroup/test1.jar**.
dependency_files	No	Array of Strings	Name of the resource package that has been uploaded to the DLI resource management system. This parameter is used to customize dependency files. Example: **myGroup/test.cvs,myGroup/test1.csv**.
tm_cus	No	Integer	Number of CUs for each TaskManager. The default value is **1**.
tm_slot_num	No	Integer	Number of slots in each TaskManager. The default value is **(parallel_number*tm_cus)/(cu_number-manager_cu_number)**.
feature	No	String	Job feature. Type of the Flink image used by a job. • **basic**: indicates that the basic Flink image provided by DLI is used. • **custom**: indicates that the user-defined Flink image is used.

Parameter	Mandatory	Type	Description
flink_version	No	String	Flink version. This parameter is valid only when **feature** is set to **basic**. You can use this parameter with the **feature** parameter to specify the version of the DLI basic Flink image used for job running.
image	No	String	Custom image. The format is **Organization name/Image name:Image version**. This parameter is valid only when **feature** is set to **custom**. You can use this parameter with the **feature** parameter to specify a user-defined Flink image for job running. For details about how to use custom images, see .
resume_checkpoint	No	Boolean	Whether the abnormal restart is recovered from the checkpoint.
resume_max_num	No	Integer	Maximum number of retry times upon exceptions. The unit is times/hour. Value range: -1 or greater than 0. The default value is **-1**, indicating that the number of times is unlimited.
checkpoint_path	No	String	Storage address of the checkpoint in the JAR file of the user. The path must be unique.
runtime_config	No	String	Customizes optimization parameters when a Flink job is running.
job_type	No	String	Job types.

Response

Table 12-20 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	String	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	Message content.
job	No	object	Information about job update. For details, see **Table 12-21**.

Table 12-21 job parameters

Parameter	Mandatory	Type	Description
update_time	No	Long	Time when a job is updated. The unit is millisecond.

Example Request

Update the Flink Jar job information. After the update, the job name is **test1**, the job execution queue is **testQueue**, and the job log function is disabled.

```
{
    "name": "test1",
    "desc": "job for test",
    "job_type": "flink_jar_job",
    "queue_name": "testQueue",
    "manager_cu_number": 1,
    "cu_number": 2,
    "parallel_number": 1,
    "log_enabled": false,
    "main_class": "org.apache.flink.examples.streaming.JavaQueueStream",
    "restart_when_exception": false,
    "entrypoint": "FemaleInfoCollec.jar",
    "dependency_jars": [
        "myGroup/test.jar",
        "myGroup/test1.jar"
    ],
    "dependency_files": [
        "myGroup/test.csv",
        "myGroup/test1.csv"
    ]
}
```

Example Response

```
{
  "is_success": true,
  "message": "The Flink job is updated successfully.",
  "job": {
     "update_time": 1516952770835
  }
}
```

Status Codes

Table 12-22 describes status codes.

Table 12-22 Status codes

Status Code	Description
200	The custom Flink job is updated successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

12.5 Running Jobs in Batches

Function

This API is used to trigger batch job running.

URI

- URI format

  POST /v1.0/{project_id}/streaming/jobs/run
- Parameter description

  Table 12-23 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 12-24 Request parameters

Parameter	Mandatory	Type	Description
job_ids	Yes	Array of Long	Batch job ID. You can obtain the job ID by calling the API for creating a job or the API for querying a job.
resume_savepoint	No	Boolean	Whether to restore a job from the latest savepoint. • If **resume_savepoint** is set to **true**, the job is restored from the latest savepoint. • If **resume_savepoint** is set to **false**, the job is started normally, not from a specific savepoint. The default value is **false**.

Response

Table 12-25 Response parameter

Parameter	Mandatory	Type	Description
Array elements	No	Array of objects	The response message returned is as follows: For details, see **Table 12-26**.

Table 12-26 Array element parameters

Parameter	Mandatory	Type	Description
is_success	No	String	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	Message content.

Example Request

Run the jobs whose IDs are **131**, **130**, **138**, and **137** and allow the jobs to be restored from their latest savepoints.

```
{
   "job_ids": [131,130,138,137],
   "resume_savepoint": true
}
```

Example Response

```
[
   {
      "is_success": "true",
      "message": "The request for submitting DLI jobs is delivered successfully."
   },
   {
      "is_success": "true",
      "message": "The request for submitting DLI jobs is delivered successfully."
   },
   {
      "is_success": "true",
      "message": "The request for submitting DLI jobs is delivered successfully."
   },
   {
      "is_success": "true",
      "message": "The request for submitting DLI jobs is delivered successfully."
   }
]
```

Status Codes

Table 12-27 describes status codes.

Table 12-27 Status codes

Status Code	Description
200	Jobs are successfully run in batches.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

12.6 Listing Jobs

Function

This API is used to list the current user's jobs. You can set the job ID as the ID and query jobs whose IDs are greater than or less than the ID. You can also query jobs in specific status, for example, in running status or other. By default, all jobs are queried.

URI

- URI format

  GET /v1.0/{project_id}/streaming/jobs
- Parameter description

  Table 12-28 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 12-29 query parameter description

Parameter	Mandatory	Type	Description
job_type	No	String	Job type. • **flink_sql_job**: Flink SQL job • **flink_opensource_sql_job**: Flink OpenSource SQL job • **flink_jar_job**: User-defined Flink job
status	No	String	Job status code. Available job statuses are as follows: • **job_init**: The job is in the draft status. • **job_submitting**: The job is being submitted. • **job_submit_fail**: The job fails to be submitted. • **job_running**: The job is running. (The billing starts. After the job is submitted, a normal result is returned.) • **job_running_exception** (The billing stops. The job stops running due to an exception.) • **job_downloading**: The job is being downloaded. • **job_idle**: The job is idle. • **job_canceling**: The job is being stopped. • **job_cancel_success**: The job has been stopped. • **job_cancel_fail**: The job fails to be stopped. • **job_savepointing**: The savepoint is being created. • **job_arrearage_stopped**: The job is stopped because the account is in arrears. (The billing ends. The job is stopped because the user account is in arrears.) • **job_arrearage_recovering**: The recharged job is being restored. (The account in arrears is recharged, and the job is being restored). • **job_finish**: The job is completed.

Parameter	Mandatory	Type	Description
queue_name	No	String	Name of a queue.
order	No	String	Sorting style of the query results. • **asc**: by time in ascending order • **desc**: by time in descending order The default value is **desc**.
limit	No	Integer	Number of returned data records. The default value is **10** and the maximum value is **100**.
name	No	String	Name of the job. Length range: 0 to 57 characters.
offset	No	Integer	Job offset.
show_detail	No	Boolean	Whether to return job details. The default value is **false**. If this parameter is set to **true**, the job details are returned. For details, see **Querying Job Details**.
user_name	No	String	Username, which can be used as a filter.
tags	No	String	Specifies a label for filtering.

Request

None

Response

Table 12-30 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	String	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_list	No	Object	Information about a job list. For details, see **Table 12-31**.

Table 12-31 job_list parameters

Parameter	Mandatory	Type	Description
total_count	No	Integer	Number of records in the query result.
jobs	No	Array of objects	Information about a job. For details, see **Table 12-32**.

Table 12-32 Jobs parameters

Parameter	Mandatory	Type	Description
job_id	No	Long	Job ID.
name	No	String	Name of the job. Length range: 0 to 57 characters.
desc	No	String	Job description. Length range: 0 to 512 characters.
user_name	No	String	Username. This parameter is valid only when **show_detail** is set to **false**.
job_type	No	String	Job type. • **flink_sql_job**: Flink SQL job • **flink_opensource_sql_job**: Flink open-source SQL job • **flink_jar_job**: User-defined Flink job
status	No	String	Job status.
status_desc	No	String	Description of job status.
create_time	No	Long	Time when a job is created.
start_time	No	Long	Time when a job is started. The value **0** indicates that the process is not started.
duration	No	Long	Running duration of a job. Unit: ms. This parameter is valid only when **show_detail** is set to **false**.
root_id	No	Long	Parent job ID. This parameter is valid only when **show_detail** is set to **false**.

Parameter	Mandatory	Type	Description
graph_editor_enabled	No	Boolean	Whether the flow diagram can be edited. Value **true** indicates that the flow diagram can be edited, and **false** indicates that the flow diagram cannot be edited.
has_savepoint	No	Boolean	Whether a job has a savepoint. Value **true** indicates that the job has a savepoint, and **false** indicates that the job does not have a savepoint.
user_id	No	String	ID of the user who creates the job. This parameter is valid only when **show_detail** is set to **true**.
project_id	No	String	ID of the project to which a job belongs. This parameter is valid only when **show_detail** is set to **true**.
sql_body	No	String	Stream SQL statement. This parameter is valid only when **show_detail** is set to **false**.
run_mode	No	String	Job running mode. The options are as follows: The value can be **shared_cluster**, **exclusive_cluster**, or **edge_node**. This parameter is valid only when **show_detail** is set to **true**. • **shared_cluster**: indicates that the job is running on a shared cluster. • **exclusive_cluster**: indicates that the job is running on an exclusive cluster. • **edge_node**: indicates that the job is running on an edge node.
job_config	No	Object	Job configuration. This parameter is valid only when **show_detail** is set to **false**. For details, see **Table 12-33**.
main_class	No	String	Main class of a JAR package. This parameter is valid only when **show_detail** is set to **false**.
entrypoint_args	No	String	Job running parameter of the JAR file. Multiple parameters are separated by spaces. This parameter is valid only when **show_detail** is set to **true**.
execution_graph	No	String	Job execution plan. This parameter is valid only when **show_detail** is set to **false**.
update_time	No	Long	Time when a job is updated. This parameter is valid only when **show_detail** is set to **false**.

Parameter	Mandatory	Type	Description
queue_name	No	String	Queue name
edge_group_ids	No	Array of Strings	List of edge computing group IDs. Use commas (,) to separate multiple IDs.
restart_times	No	Integer	Number of restart times
savepoint_path	No	String	Path for storing manually generated checkpoints

Table 12-33 job_config parameters

Parameter	Mandatory	Type	Description
checkpoint_enabled	No	Boolean	Whether to enable the automatic job snapshot function. ● **true**: The automatic job snapshot function is enabled. ● **false**: The automatic job snapshot function is disabled. The default value is **false**.
checkpoint_mode	No	String	Snapshot mode. There are two options: ● **exactly_once**: indicates that data is processed only once. ● **at_least_once**: indicates that data is processed at least once. The default value is **exactly_once**.
checkpoint_interval	No	Integer	Snapshot interval. The unit is second. The default value is **10**.
log_enabled	No	Boolean	Whether to enable the log storage function. The default value is **false**.
obs_bucket	No	String	Name of an OBS bucket.
smn_topic	No	String	SMN topic name. If a job fails, the system will send a message to users subscribed to the SMN topic.
root_id	No	Integer	Parent job ID.

Parameter	Mandatory	Type	Description
edge_group_ids	No	Array of Strings	List of edge computing group IDs. Use commas (,) to separate multiple IDs.
manager_cu_number	No	Integer	Number of CUs of the management unit. The default value is **1**.
cu_number	No	Integer	Number of CUs selected for a job. This parameter is valid only when **show_detail** is set to **true**. ● Minimum value: **2** ● Maximum value: **400** The default value is **2**.
parallel_number	No	Integer	Number of concurrent jobs set by a user. This parameter is valid only when **show_detail** is set to **true**. ● Minimum value: **1** ● Maximum value: **2000** The default value is **1**.
restart_when_exception	No	Boolean	Whether to enable the function of restart upon exceptions.
idle_state_retention	No	Integer	Expiration time.
udf_jar_url	No	String	Name of the package that has been uploaded to the DLI resource management system. The **UDF Jar** file of the SQL job is uploaded through this parameter.
dirty_data_strategy	No	String	Dirty data policy of a job. ● **2:obsDir**: Save. **obsDir** specifies the path for storing dirty data. ● **1**: Trigger a job exception ● **0**: Ignore
entrypoint	No	String	Name of the package that has been uploaded to the DLI resource management system. This parameter is used to customize the JAR file where the job main class is located.
dependency_jars	No	Array of Strings	Name of the package that has been uploaded to the DLI resource management system. This parameter is used to customize other dependency packages.

Parameter	Mandatory	Type	Description
dependency_files	No	Array of Strings	Name of the resource package that has been uploaded to the DLI resource management system. This parameter is used to customize dependency files.
executor_number	No	Integer	Number of compute nodes in a job.
executor_cu_number	No	Integer	Number of CUs in a compute node.
resume_checkpoint	No	Boolean	Whether to restore data from the latest checkpoint when the system automatically restarts upon an exception. The default value is **false**.
runtime_config	No	String	Customizes optimization parameters when a Flink job is running.
graph_editor_enabled	No	Boolean	Whether to enable flow diagram editing. The default value is **false**.
graph_editor_data	No	String	Edited stream graph data. The default value is **null**.
resume_max_num	No	Integer	Maximum retry attempts. **–1** indicates there is no upper limit.
checkpoint_path	No	String	Path for saving the checkpoint.
config_url	No	String	OBS path of the **config** package uploaded by the user.
tm_cus	No	int	Number of CUs per TaskManager node.
tm_slot_num	No	int	Number of slots per TaskManager node.
image	No	String	Custom image. The format is **Organization name/Image name:Image version**. This parameter is valid only when **feature** is set to **custom**. You can use this parameter with the **feature** parameter to specify a user-defined Flink image for job running.
feature	No	String	User-defined job feature. Type of the Flink image used by a job. • **basic**: indicates that the basic Flink image provided by DLI is used. • **custom**: indicates that the user-defined Flink image is used.

Parameter	Mandatory	Type	Description
flink_version	No	String	Flink version. This parameter is valid only when **feature** is set to **basic**. You can use this parameter with the **feature** parameter to specify the version of the DLI basic Flink image used for job running.
operator_config	No	String	Operator's parallelism degree. The operator ID and degree of parallelism are displayed in JSON format.
static_estimator_config	No	String	Estimation of static flow diagram resources.
real_cu_number	No	Integer	Number of actually used CUs. The default value is **0**, indicating that the value of **cu_number** is used.

Example Request

None

Example Response

```
{
    "is_success": "true",
    "message": "Querying of the job list succeeds.",
    "job_list": {
        "total_count": 26,
        "jobs": [
            {
                "job_id": 146,
                "name": "aaaaa",
                "desc": "",
                "user_name": "",
                "job_type": "flink_sql_job",
                "status": "job_init",
                "status_desc": "",
                "create_time": 1578892414688,
                "duration": 0,
                "root_id": -1,
                "graph_editor_enabled": false,
                "has_savepoint": false
            }
        ]
    }
}
```

Status Codes

Table 12-34 describes the status code.

Table 12-34 Status codes

Status Code	Description
200	Job list query succeeds.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

12.7 Querying Job Details

Function

This API is used to query details of a job.

URI

- URI format
  GET /v1.0/{project_id}/streaming/jobs/{job_id}
- Parameter description

  Table 12-35 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
job_id	Yes	String	Job ID.

Request

None

Response

Table 12-36 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	String	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_detail	No	Object	Job details. For details, see **Table 12-37**.

Table 12-37 job_detail parameters

Parameter	Mandatory	Type	Description
job_id	No	Long	Job ID.
name	No	String	Name of the job. Length range: 0 to 57 characters.
desc	No	String	Job description. Length range: 0 to 512 characters.
job_type	No	String	Job type. • **flink_sql_job**: Flink SQL job • **flink_opensource_sql_job**: Flink OpenSource SQL job • **flink_jar_job**: User-defined Flink job

Parameter	Mandatory	Type	Description
status	No	String	Job status. Available job statuses are as follows: • **job_init**: The job is in the draft status. • **job_submitting**: The job is being submitted. • **job_submit_fail**: The job fails to be submitted. • **job_running**: The job is running. (The billing starts. After the job is submitted, a normal result is returned.) • **job_running_exception** (The billing stops. The job stops running due to an exception.) • **job_downloading**: The job is being downloaded. • **job_idle**: The job is idle. • **job_canceling**: The job is being stopped. • **job_cancel_success**: The job has been stopped. • **job_cancel_fail**: The job fails to be stopped. • **job_savepointing**: The savepoint is being created. • **job_arrearage_stopped**: The job is stopped because the account is in arrears. (The billing ends. The job is stopped because the user account is in arrears.) • **job_arrearage_recovering**: The recharged job is being restored. (The account in arrears is recharged, and the job is being restored). • **job_finish**: The job is completed.
status_desc	No	String	Description of job status.
create_time	No	Long	Time when a job is created.
start_time	No	Long	Time when a job is started.
user_id	No	String	ID of the user who creates the job.
queue_name	No	String	Name of a queue. Length range: 1 to 128 characters.
project_id	No	String	ID of the project to which a job belongs.
sql_body	No	String	Stream SQL statement.

Parameter	Mandatory	Type	Description
savepoint_path	No	String	Path for storing manually generated checkpoints.
run_mode	No	String	Job running mode. The options are as follows: • **shared_cluster**: indicates that the job is running on a shared cluster. • **exclusive_cluster**: indicates that the job is running on an exclusive cluster. • **edge_node**: indicates that the job is running on an edge node.
job_config	No	Object	Job configurations. Refer to **Table 12-38** for details.
main_class	No	String	Main class of a JAR package, for example, **org.apache.spark.examples.streaming.JavaQueueStream**.
entrypoint_args	No	String	Running parameter of a JAR package job. Multiple parameters are separated by spaces.
execution_graph	No	String	Job execution plan.
update_time	No	Long	Time when a job is updated.
user_name	No	String	Username. This parameter is valid only when **show_detail** is set to **false**.
duration	No	Long	Running duration of a job. Unit: ms. This parameter is valid only when **show_detail** is set to **false**.
root_id	No	Long	Parent job ID. This parameter is valid only when **show_detail** is set to **false**.
graph_editor_enabled	No	Boolean	Whether the stream graph of a job can be edited. Value **true** indicates that the flow diagram can be edited, and **false** indicates that the flow diagram cannot be edited.
has_savepoint	No	Boolean	Whether savepointing is enabled for the job. Value **true** indicates that the job has a savepoint, and **false** indicates that the job does not have a savepoint.
edge_group_ids	No	Array of Strings	List of edge computing group IDs. Use commas (,) to separate multiple IDs.
restart_times	No	Integer	Number of restart times

Table 12-38 job_config parameters

Parameter	Mandatory	Type	Description
checkpoint_enabled	No	Boolean	Whether to enable the automatic job snapshot function. • **true**: The automatic job snapshot function is enabled. • **false**: The automatic job snapshot function is disabled. The default value is **false**.
checkpoint_interval	No	Integer	Snapshot interval. The unit is second. The default value is **10**.
checkpoint_mode	No	String	Snapshot mode. There are two options: • **exactly_once**: indicates that data is processed only once. • **at_least_once**: indicates that data is processed at least once. The default value is **exactly_once**.
log_enabled	No	Boolean	Whether to enable the log storage function. The default value is **false**.
obs_bucket	No	String	Name of an OBS bucket.
root_id	No	Integer	Parent job ID.
edge_group_ids	No	Array of Strings	List of edge computing group IDs. Use commas (,) to separate multiple IDs.
manager_cu_number	No	Integer	Number of CUs of the management unit. The default value is **1**.
graph_editor_enabled	No	Boolean	Whether to enable flow diagram editing. The default value is **false**.
graph_editor_data	No	String	Data of flow diagram editing. The default value is **null**.
executor_number	No	Integer	Number of compute nodes in a job.
executor_cu_number	No	Integer	Number of CUs in a compute node.

Parameter	Mandatory	Type	Description
cu_number	No	Integer	Number of CUs selected for a job. This parameter is valid only when **show_detail** is set to **true**. ● Minimum value: **2** ● Maximum value: **400** The default value is **2**.
parallel_number	No	Integer	Number of concurrent jobs set by a user. This parameter is valid only when **show_detail** is set to **true**. ● Minimum value: **1** ● Maximum value: **2000** The default value is **1**.
smn_topic	No	String	SMN topic name. If a job fails, the system will send a message to users subscribed to this SMN topic.
restart_when_exception	No	Boolean	Whether to enable the function of restart upon exceptions.
resume_checkpoint	No	Boolean	Whether to restore data from the latest checkpoint when the system automatically restarts upon an exception. The default value is **false**.
resume_max_num	No	Integer	Maximum retry attempts. **–1** indicates there is no upper limit.
checkpoint_path	No	String	Path for saving the checkpoint.
idle_state_retention	No	Integer	Expiration time.
config_url	No	String	OBS path of the **config** package uploaded by the user.
udf_jar_url	No	String	Name of the package that has been uploaded to the DLI resource management system. The **UDF Jar** file of the SQL job is uploaded through this parameter.
dirty_data_strategy	No	String	Dirty data policy of a job. ● **2:obsDir**: Save. **obsDir** specifies the path for storing dirty data. ● **1**: Trigger a job exception ● **0**: Ignore

Parameter	Mandatory	Type	Description
entrypoint	No	String	Name of the package that has been uploaded to the DLI resource management system. This parameter is used to customize the JAR file where the job main class is located.
dependency_jars	No	Array of Strings	Name of the package that has been uploaded to the DLI resource management system. This parameter is used to customize other dependency packages.
dependency_files	No	Array of Strings	Name of the resource package that has been uploaded to the DLI resource management system. This parameter is used to customize dependency files.
tm_cus	No	int	Number of CUs per TaskManager node.
tm_slot_num	No	int	Number of slots per TaskManager node.
image	No	String	Custom image. The format is **Organization name/Image name:Image version**. This parameter is valid only when **feature** is set to **custom**. You can use this parameter with the **feature** parameter to specify a user-defined Flink image for job running.
feature	No	String	User-defined job feature. Type of the Flink image used by a job. • **basic**: indicates that the basic Flink image provided by DLI is used. • **custom**: indicates that the user-defined Flink image is used.
flink_version	No	String	Flink version. This parameter is valid only when **feature** is set to **basic**. You can use this parameter with the **feature** parameter to specify the version of the DLI basic Flink image used for job running.
operator_config	No	String	Operator's parallelism degree. The operator ID and degree of parallelism are displayed in JSON format.
static_estimator_config	No	String	Estimation of static flow diagram resources.
runtime_config	No	String	Customizes optimization parameters when a Flink job is running.

Parameter	Mandatory	Type	Description
real_cu_number	No	Integer	Number of actually used CUs. The default value is **0**, indicating that the value of **cu_number** is used.

Example Request

None

Example Response

- The following example takes the **flink_jar_job** type as an example:


```
{
    "is_success": "true",
    "message": "Job detail query succeeds.",
    "job_detail": {
        "job_id": 104,
        "user_id": "011c99a26ae84a1bb963a75e7637d3fd",
        "queue_name": "flinktest",
        "project_id": "330e068af1334c9782f4226acc00a2e2",
        "name": "jptest",
        "desc": "",
        "sql_body": "",
        "run_mode": "exclusive_cluster",
        "job_type": "flink_jar_job",
        "job_config": {
            "checkpoint_enabled": false,
            "checkpoint_interval": 10,
            "checkpoint_mode": "exactly_once",
            "log_enabled": false,
            "obs_bucket": null,
            "root_id": -1,
            "edge_group_ids": null,
            "graph_editor_enabled": false,
            "graph_editor_data": "",
            "manager_cu_number": 1,
            "executor_number": null,
            "executor_cu_number": null,
            "cu_number": 2,
            "parallel_number": 1,
            "smn_topic": null,
            "restart_when_exception": false,
            "idle_state_retention": 3600,
            "config_url": null,
            "udf_jar_url": null,
            "dirty_data_strategy": null,
            "entrypoint": "FemaleInfoCollection.jar",
            "dependency_jars": [
                "FemaleInfoCollection.jar",
                "ObsBatchTest.jar"
            ],
            "dependency_files": [
                "FemaleInfoCollection.jar",
                "ReadFromResource"
            ]
        },
        "main_class": null,
        "entrypoint_args": null,
        "execution_graph": null,
        "status": "job_init",
```



```
        "status_desc": "",
        "create_time": 1578466221525,
        "update_time": 1578467395713,
        "start_time": null
    }
}
```

- The following example takes the **flink_opensource_sql_job** type as an example:

```
{
    "is_success": "true",
    "message": "The job information query succeeds.",
    "job_detail": {
        "job_type": "flink_opensource_sql_job",
        "status_desc": "",
        "create_time": 1637632872828,
        "sql_body": "xxx",
        "savepoint_path": null,
        "main_class": null,
        "queue_name": "xie_container_general",
        "execution_graph": "xxx",
        "start_time": 1638433497621,
        "update_time": 1638449337993,
        "job_config": {
            "checkpoint_enabled": true,
            "checkpoint_interval": 600,
            "checkpoint_mode": "exactly_once",
            "log_enabled": true,
            "obs_bucket": "dli-test",
            "root_id": -1,
            "edge_group_ids": null,
            "graph_editor_enabled": false,
            "graph_editor_data": "",
            "manager_cu_number": 1,
            "executor_number": null,
            "executor_cu_number": null,
            "cu_number": 2,
            "parallel_number": 3,
            "smn_topic": "",
            "restart_when_exception": true,
            "resume_checkpoint": true,
            "resume_max_num": -1,
            "checkpoint_path": null,
            "idle_state_retention": 3600,
            "config_url": null,
            "udf_jar_url": "test/flink_test-1.0-SNAPSHOT-jar-with-dependencies.jar",
            "dirty_data_strategy": "0",
            "entrypoint": "test/flink_test-1.0-SNAPSHOT-jar-with-dependencies.jar",
            "dependency_jars": null,
            "dependency_files": null,
            "tm_cus": 1,
            "tm_slot_num": 3,
            "image": null,
            "feature": null,
            "flink_version": null,
            "operator_config": "xxx",
            "static_estimator_config": "xxx",
            "runtime_config": null
        },
        "user_id": "xxx",
        "project_id": "xxx",
        "run_mode": "exclusive_cluster",
        "job_id": 90634,
        "name": "test_guoquan",
        "desc": "",
        "entrypoint_args": null,
        "status": "job_cancel_success"
    }
}
```


Status Codes

Table 12-39 describes the status code.

Table 12-39 Status codes

Status Code	Description
200	Querying details of a job succeeds.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

12.8 Querying the Job Execution Plan

Function

This API is used to query a job execution plan.

URI

- URI format

  GET /v1.0/{project_id}/streaming/jobs/{job_id}/execute-graph
- Parameter description

  Table 12-40 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
job_id	Yes	Long	Job ID.

Request

None

Response

Table 12-41 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	String	Whether the request is successful.
message	No	String	Message content.
execute_graph	No	Object	Response parameter for querying a job plan. For details, see **Table 12-42**.

Table 12-42 execute_graph parameters

Parameter	Mandatory	Type	Description
jid	No	String	ID of a Flink job.
name	No	String	Name of a Flink job.
isStoppable	No	Boolean	Whether a job can be stopped.
state	No	String	Execution status of a job.
start-time	No	Long	Time when a job is started.
end-time	No	Long	Time when a job is stopped.
duration	No	Long	Running duration of a job.

Example Request

None

Example Response

```
{
    "is_success": "true",
  "message": "Querying the job execution graph succeeds.",
    "execute_graph": {
        "jid": "4e966f43f2c90b0e1bf3188ecf55504b",
        "name": "",
        "isStoppable": false,
        "state": "RUNNING",
        "start-time": 1578904488436,
        "end-time": -1,
        "duration": 516274
    }
}
```


Status Codes

Table 12-43 Status codes

Status Code	Description
200	Querying the job execution plan succeeds.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

12.9 Stopping Jobs in Batches

Function

This API is used to stop running jobs in batches.

URI

- URI format

  POST /v1.0/{project_id}/streaming/jobs/stop
- Parameter description

  Table 12-44 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 12-45 Request parameters

Parameter	Mandatory	Type	Description
job_ids	Yes	Array of Long	Job ID.

Parameter	Mandatory	Type	Description
trigger_savepoint	No	Boolean	Whether to create a savepoint for a job to store the job status information before stopping it. The data type is Boolean. • If this parameter is set to **true**, a savepoint is created. • If this parameter is set to **false**, no savepoint is created. The default value is **false**.

Response

Table 12-46 Response parameters

Parameter	Mandatory	Type	Description
Array elements	No	Array of objects	The response message returned is as follows: For details, see **Table 12-47**.

Table 12-47 Array element parameters

Parameter	Mandatory	Type	Description
is_success	No	String	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	Message content

Example Request

Stop the jobs whose IDs are **128** and **137**.

```
{
  "job_ids": [128, 137],
  "trigger_savepoint": false
}
```

Example Response

```
[{"is_success":"true",
"message": "The request for stopping DLI jobs is delivered successfully."}]
```


Status Codes

Table 12-48 describes status codes.

Table 12-48 Status codes

Status Code	Description
200	The request of stopping a job is sent successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

12.10 Deleting a Job

Function

This API is used to delete a Flink job at any state.

NOTE

The job records will not be deleted.

URI

- URI format

  DELETE /v1.0/{project_id}/streaming/jobs/{job_id}
- Parameter description

Table 12-49 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
job_id	Yes	Long	Job ID.

Request

None

Response

Table 12-50 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	String	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Example Request

None

Example Response

```
{
    "is_success": "true",
    "message": "The job is deleted successfully.",
}
```

Status Codes

Table 12-51 describes status codes.

Table 12-51 Status codes

Status Code	Description
200	The job is deleted successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

12.11 Deleting Jobs in Batches

Function

This API is used to batch delete jobs at any state.

URI

- URI format

  POST /v1.0/{project_id}/streaming/jobs/delete

- Parameter description

  Table 12-52 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 12-53 Request parameter

Parameter	Mandatory	Type	Description
job_ids	Yes	[Long]	Job ID.

Response

Table 12-54 Response parameters

Parameter	Mandatory	Type	Description
Array elements	No	Array of Objects	Returned response message. For details, see **Table 12-55**.

Table 12-55 Array element parameters

Parameter	Mandatory	Type	Description
is_success	No	String	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	Message content.

Example Request

Delete the jobs whose IDs are **12** and **232**.

```
{
  "job_ids":[12,232]
}
```

Example Response

```
[{
    "is_success": "true",
    "message": "The job is deleted successfully.",
}]
```

Status Codes

Table 12-56 describes status codes.

Table 12-56 Status codes

Status Code	Description
200	The job is deleted successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

12.12 Exporting a Flink Job

Function

This API is used to export Flink job data.

URI

- URI format

  POST /v1.0/{project_id}/streaming/jobs/export
- Parameter description

Table 12-57 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 12-58 Request parameters

Parameter	Mandatory	Type	Description
obs_dir	Yes	String	OBS path for storing exported job files.
is_selected	Yes	Boolean	Whether to export a specified job.
job_selected	No	Array of Longs	This parameter indicates the ID set of jobs to be exported if **is_selected** is set to **true**. **NOTE** This parameter is mandatory when **is_selected** is set to **true**.

Response

Table 12-59 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	String	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
zip_file	No	Array of Strings	Name of the ZIP package containing exported jobs. The ZIP package is stored on OBS.

Example Request

Export the job whose ID is **100** to OBS.

```
{
   "obs_dir": "obs-test",
```



```
    "is_selected": true,
    "job_selected": [100]
}
```

Example Response

```
{
    "is_success": true,
    "message": "The job is exported successfully.",
    "zip_file": ["obs-test/aggregate_1582677879475.zip"]
}
```

Status Codes

Table 12-60 describes status codes.

Table 12-60 Status codes

Status Code	Description
200	The job is exported successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

12.13 Importing a Flink Job

Function

This API is used to import Flink job data.

URI

- URI format

  POST /v1.0/{project_id}/streaming/jobs/import
- Parameter description

  Table 12-61 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 12-62 Request parameters

Parameter	Mandatory	Type	Description
zip_file	Yes	String	Path of the job ZIP file imported from OBS. You can enter a folder path to import all ZIP files in the folder. **NOTE** The folder can contain only **.zip** files.
is_cover	No	Boolean	Whether to overwrite an existing job if the name of the imported job is the same as that of the existing job in the service.

Response

Table 12-63 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	String	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_mapping	No	Array of objects	Information about the imported job. For details, see **Table 12-64**.

Table 12-64 job_mapping parameter description

Parameter	Mandatory	Type	Description
old_job_id	No	Long	ID of a job before being imported.
new_job_id	No	Long	ID of a job after being imported. If **is_cover** is set to **false** and a job with the same name exists in the service, the returned value of this parameter is **-1**.
remark	No	String	Results about an imported job.

Example Request

Whether to overwrite the existing job if the name of the imported job is the same as that of an existing job when Flink job data is imported from OBS.

```
{
   "zip_file": "test/ggregate_1582677879475.zip",
   "is_cover": true
}
```

Example Response

```
{
   "is_success": true,
   "message": "The job is imported successfully.",
   "job_mapping": [
      {
         "old_job_id": "100",
         "new_job_id": "200",
         "remark": "Job successfully created"
      }
   ]
}
```

Status Codes

Table 12-65 describes status codes.

Table 12-65 Status codes

Status Code	Description
200	The job is imported successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

12.14 Generating a Static Stream Graph for a Flink SQL Job

Function

This API is used to generate a static stream graph for a Flink SQL job.

Flink 1.15 does not support the generation of static stream graphs.

URI

- URI format

  POST /v3/{*project_id*}/streaming/jobs/{*job_id*}/gen-graph

- Parameter description

Table 12-66 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 12-67 Request parameters

Parameter	Mandatory	Type	Description
sql_body	Yes	String	SQL
cu_number	No	Integer	Total number of CUs used by the job configured on the job editing page, which should match the actual number of CUs used. The actual number of CUs used is determined by the number of parallel operators. Total number of CUs used by the job = Number of manager CUs + (Total number of concurrent operators / Number of slots of a TaskManager) x Number of TaskManager CUs
manager_cu_number	No	Integer	Number of CUs of the management unit.
parallel_number	No	Integer	Maximum degree of parallelism. Concurrent tasks of each job operator. Appropriately increasing the value will improve the overall computing performance of a job. Considering switchover overheads due to increasing threads, the maximum value is four times the number of CUs. One to two times the number of CUs is the optimal.
tm_cus	No	Integer	Number of CUs in a taskManager.
tm_slot_num	No	Integer	Number of slots in a taskManager.

Parameter	Mandatory	Type	Description
operator_config	No	String	Operator configurations. You can call this API to obtain the operator ID. That is, the ID in **operator_list** contained in **stream_graph** in the response message is the operator ID.
static_estimator	No	Boolean	Whether to estimate static resources. If this parameter is set to **true**, resources used by the job are estimated based on the operator ID and traffic.
static_estimator_config	No	String	Traffic or hit ratio of each operator, which is a string in JSON format. This parameter is mandatory when **static_estimator** is set to **true**. During the configuration, the operator ID and operator traffic configuration are required. • You can call this API to obtain the operator ID. That is, the ID in **operator_list** contained in **stream_graph** in the response message is the operator ID. • The operator traffic is estimated based on the actual service conditions.
job_type	No	String	Job types. Only **flink_opensource_sql_job job** is supported.
graph_type	No	String	Stream graph type. Currently, the following two types of stream graphs are supported: • **simple_graph**: Simplified stream graph • **job_graph**: Static stream graph
flink_version	No	String	Flink version. Currently, only 1.10 and 1.12 are supported.

Response

Table 12-68 Response parameters

Parameter	Mandatory	Type	Description
is_success	Yes	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	Yes	String	System prompt. If execution succeeds, the message may be left blank.
error_code	Yes	String	Error codes.
stream_graph	Yes	String	Description of a static stream graph.

Example Request

Generate a static stream graph for a Flink SQL job.

```
{
  "job_type": "flink_opensource_sql_job",
  "graph_type": "job_graph",
  "sql_body": "create table orders(\r\n  name string,\r\n  num int\r\n) with (\r\n  'connector' = 'datagen',\r
\n  'rows-per-second' = '1', --Generates a data record per second.\r\n  'fields.name.kind' = 'random', --
Specify a random generator for the user_id field.\r\n  'fields.name.length' = '5' --Limit the length of user_id
to 3.\r\n);\r\n \r\nCREATE TABLE sink_table (\r\n  name string,\r\n  num int\r\n) WITH (\r\n   'connector' =
'print'\r\n);\r\nINSERT into sink_table SELECT * FROM orders;",
  "cu_number": 2,
  "manager_cu_number": 1,
  "parallel_number": 2,
  "tm_cus": 1,
  "tm_slot_num": 0,
  "operator_config": "",
  "static_estimator": true,
  "flink_version": "1.12",
  "static_estimator_config": "{\"operator_list\":[{\"id
\":\"0a448493b4782967b150582570326227\",\"output_rate\":1000},{\"id
\":\"bc764cd8ddf7a0cff126f51c16239658\",\"output_rate\":1000}]}"
}
```

Example Response

```
{
    "message": "",
    "is_success": true,
    "error_code": "",
    "stream_graph": "{\n  \"jid\" : \"44334c4259f6714bddef1ac525364052\",\n  \"name\" :
\"InternalJob_1715392878428\",\n  \"nodes\" : [ {\n    \"id\" :
\"0a448493b4782967b150582570326227\",\n    \"parallelism\" : 1,\n    \"operator\" : \"\",\n
\"operator_strategy\" : \"\",\n    \"description\" : \"Sink:
Sink(table=[default_catalog.default_database.sink_table], fields=[name, num])\",\n    \"chain_operators_id
\" : [ \"0a448493b4782967b150582570326227\" ],\n    \"inputs\" : [ {\n      \"num\" : 0,\n      \"id\" :
\"bc764cd8ddf7a0cff126f51c16239658\",\n      \"ship_strategy\" : \"FORWARD\",\n      \"exchange\" :
\"pipelined_bounded\"\n    } ],\n    \"optimizer_properties\" : {}\n  }, {\n    \"id\" :
\"bc764cd8ddf7a0cff126f51c16239658\",\n    \"parallelism\" : 2,\n    \"operator\" : \"\",\n
\"operator_strategy\" : \"\",\n    \"description\" : \"Source: TableSourceScan(table=[[default_catalog,
```


default_database, orders]], fields=[name, num])\",\n \"chain_operators_id\" :
[\"bc764cd8ddf7a0cff126f51c16239658\"],\n \"optimizer_properties\" : {}\n }],\n \"operator_list\" :
[{\n \"id\" : \"0a448493b4782967b150582570326227\",\n \"name\" : \"Sink:
Sink(table=[default_catalog.default_database.sink_table], fields=[name, num])\",\n \"type\" : \"Sink\",\n
\"contents\" : \"Sink(table=[default_catalog.default_database.sink_table], fields=[name, num])\",\n
\"parallelism\" : 1,\n \"tags\" : \"[SINK]\",\n \"input_operators_id\" :
[\"bc764cd8ddf7a0cff126f51c16239658\"]\n }, {\n \"id\" : \"bc764cd8ddf7a0cff126f51c16239658\",\n
\"name\" : \"Source: TableSourceScan(table=[[default_catalog, default_database, orders]], fields=[name,
num])\",\n \"type\" : \"Source\",\n \"contents\" : \"TableSourceScan(table=[[default_catalog,
default_database, orders]], fields=[name, num])\",\n \"parallelism\" : 2,\n \"tags\" : \"[PROCESS, UDF]
\",\n \"input_operators_id\" : [ ]\n }]\n}"
}

To make it easier to view the response information, we format **stream_graph** as follows:


```
  "jid": "65b6a7b0c1ad95b1722a92b49d2f6eba",
  "name": "InternalJob_1715392245413",
  "nodes": [
    {
      "id": "0a448493b4782967b150582570326227",
      "parallelism": 1,
      "operator": "",
      "operator_strategy": "",
      "description": "Sink: Sink(table=[default_catalog.default_database.sink_table], fields=[name, num])",
      "chain_operators_id": [
        "0a448493b4782967b150582570326227"
      ],
      "inputs": [
        {
          "num": 0,
          "id": "bc764cd8ddf7a0cff126f51c16239658",
          "ship_strategy": "FORWARD",
          "exchange": "pipelined_bounded"
        }
      ],
      "optimizer_properties": {

      }
    },
    {
      "id": "bc764cd8ddf7a0cff126f51c16239658",
      "parallelism": 2,
      "operator": "",
      "operator_strategy": "",
      "description": "Source: TableSourceScan(table=[[default_catalog, default_database, orders]],
fields=[name, num])",
      "chain_operators_id": [
        "bc764cd8ddf7a0cff126f51c16239658"
      ],
      "optimizer_properties": {

      }
    }
  ],
  "operator_list": [
    {
      "id": "0a448493b4782967b150582570326227",
      "name": "Sink: Sink(table=[default_catalog.default_database.sink_table], fields=[name, num])",
      "type": "Sink",
      "contents": "Sink(table=[default_catalog.default_database.sink_table], fields=[name, num])",
      "parallelism": 1,
      "tags": "[SINK]",
      "input_operators_id": [
        "bc764cd8ddf7a0cff126f51c16239658"
      ]
    },
    {
      "id": "bc764cd8ddf7a0cff126f51c16239658",
      "name": "Source: TableSourceScan(table=[[default_catalog, default_database, orders]],
```



```
fields=[name, num])",
          "type": "Source",
          "contents": "TableSourceScan(table=[[default_catalog, default_database, orders]], fields=[name,
num])",
          "parallelism": 2,
          "tags": "[PROCESS, UDF]",
          "input_operators_id": [

          ]
       }
   ]
}
```


Status Codes

Table 12-69 describes status codes.

Table 12-69 Status codes

Status Code	Description
200	The operation is successful.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

13 APIs Related to Flink Job Templates

13.1 Creating a Template

Function

This API is used to create a user template for the DLI service. A maximum of 100 user templates can be created.

URI

- URI format

  POST /v1.0/{project_id}/streaming/job-templates

- Parameter description

Table 13-1 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 13-2 Request parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Template name. The value can contain 1 to 64 characters.

Parameter	Mandatory	Type	Description
desc	No	String	Template description. Length range: 0 to 512 characters.
sql_body	No	String	Stream SQL statement, which includes at least the following three parts: source, query, and sink. Length range: 0 to 2,048 characters.
tags	No	Array of objects	Label of a Flink job template. For details, see **Table 13-3**.
job_type	No	String	Flink job template type. The default value is **flink_sql_job**. You can only set this parameter to **flink_sql_job** or **flink_opensource_sql_job**.

Table 13-3 tags parameter

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key. **NOTE** A tag key can contain a maximum of 36 characters. Special characters (=*<>|) are not allowed, and the key cannot start with a space.
value	Yes	String	Tag key. **NOTE** A tag value can contain a maximum of 43 characters. Special characters (=*<>|) are not allowed, and the value cannot start with a space.

Response

Table 13-4 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successful.
message	No	String	Message content.
template	No	Object	Information about job update. For details, see **Table 13-5**.

Table 13-5 template parameters

Parameter	Mandatory	Type	Description
template_id	No	Long	Template ID.
name	No	String	Template name.
desc	No	String	Template description.
create_time	No	Long	Time when the template is created.
job_type	No	String	Job template type

Example Request

Create a job template named **simple_stream_sql**.

```
{
    "name": "simple_stream_sql",
    "desc": "Example of quick start",
    "sql_body": "select * from source_table",
    "job_type": "flink_sql_job"
}
```

Example Response

```
{
    "is_success": true,
    "message": "A template is created successfully.",
    "template": {
        "template_id": 0,
        "name": "IoT_example",
       "desc": "Example of quick start",
        "create_time": 1516952710040,
        "job_type": "flink_opensource_sql_job"
    }
}
```

Status Codes

Table 13-6 describes status codes.

Table 13-6 Status codes

Status Code	Description
200	A template is created successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

13.2 Updating a Template

Function

This API is used to update existing templates in DLI.

URI

- URI format

  PUT /v1.0/{project_id}/streaming/job-templates/{template_id}
- Parameter description

  Table 13-7 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
template_id	Yes	String	Template ID.

Request

Table 13-8 Request parameters

Parameter	Mandatory	Type	Description
name	No	String	Template name. Length range: 0 to 57 characters.
desc	No	String	Template description. Length range: 0 to 512 characters.
sql_body	No	String	Stream SQL statement, which includes at least the following three parts: source, query, and sink. Length range: 0 to 1024 x 1024 characters.

Response

Table 13-9 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Example Request

Update job template information, including the template name, template description, and template SQL statements.

```
{
    "name": "simple_stream_sql",
    "desc": "Example of quick start",
    "sql_body": "select * from source_table"
}
```

Example Response

```
{
    "is_success": "true",
    "message": "The template is updated successfully.",
}
```

Status Codes

Table 13-10 describes status codes.

Table 13-10 Status codes

Status Code	Description
200	A template is updated successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

13.3 Deleting a Template

Function

This API is used to delete a template. A template used by jobs can also be deleted.

URI

- URI format

  DELETE /v1.0/{project_id}/streaming/job-templates/{template_id}

- Parameter description

  Table 13-11 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
template_id	Yes	String	Template ID.

Request

None

Response

- Parameter description

  Table 13-12 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the response is successful. Value **true** indicates success.
message	No	String	Message content.
template	No	Object	Information about the template to be deleted. For details, see **Table 13-13**.

Table 13-13 template parameters

Parameter	Mandatory	Type	Description
template_id	No	Long	Template ID.

Example Request

None

Example Response

```
{
    "is_success": "true",
    "message": "The template is deleted successfully.",
    "template": {
        "template_id": 2
    }
}
```

Status Codes

Table 13-14 describes status codes.

Table 13-14 Status codes

Status Code	Description
200	A template is deleted successfully.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

13.4 Listing Templates

Function

This API is used to list job templates. Currently, only custom templates can be queried.

URI

- URI format

  GET /v1.0/{project_id}/streaming/job-templates

- Parameter description

Table 13-15 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 13-16 query parameter description

Parameter	Mandatory	Type	Description
name	No	String	Template name. Fuzzy query by name is supported.
tags	No	String	List of tag names. The value is **k=v** for a single tag. Multiple tags are separated by commas (,). Example: **tag1=v1,tag2=v2**.
offset	No	Long	Job offset.
limit	No	Integer	Number of returned data records. The default value is **10**.
order	No	String	Sorting style of the query results. • **asc**: The query results are displayed in ascending order. • **desc**: The query results are displayed in the descending order. The default value is **desc**.

Request

None

Response

Table 13-17 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successful.
message	No	String	Message content.

Parameter	Mandatory	Type	Description
template_list	No	Object	Information about the template list. For details, see **Table 13-18**.

Table 13-18 template_list parameters

Parameter	Mandatory	Type	Description
total_count	No	Integer	Total number of templates.
templates	No	Array of objects	Detailed information about a template. For details, see **Table 13-19**.

Table 13-19 templates parameters

Parameter	Mandatory	Type	Description
template_id	No	Integer	Template ID.
name	No	String	Template name.
desc	No	String	Template description.
create_time	No	Long	Time when the template is created.
update_time	No	Long	Time when the template is updated.
sql_body	No	String	Stream SQL statement. Contains at least the **source**, **query**, and **sink** parts.
job_type	No	String	Job template type.

Example Request

None

Example Response

```
{
    "is_success": "true",
    "message": "The template list is obtained successfully.",
    "template_list": {
        "total_count": 2,
        "templates": [
            {
                "template_id": 2,
                "name": "updatetest",
```



```
            "desc": "Example of quick start",
            "create_time": 1578748092000,
            "update_time": 1578748092000,
            "sql_body": "select * from source_table",
            "job_type": "flink_sql_job"
        },
        {
            "template_id": 1,
            "name": "we",
            "desc": "qwe",
            "create_time": 1577951045000,
            "update_time": 1577951045000,
            "sql_body": ""
        }
    ]
  }
}
```


Status Codes

Table 13-20 describes status codes.

Table 13-20 Status codes

Status Code	Description
200	Template list query succeeds.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

14 Spark Job-related APIs

14.1 Creating a Batch Processing Job

Function

This API is used to create a batch processing job in a queue.

URI

- URI format
  POST /v2.0/{project_id}/batches
- Parameter description

Table 14-1 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 14-2 Request parameters

Parameter	Mandatory	Type	Description
file	Yes	String	Name of the package that is of the JAR or pyFile type and has been uploaded to the DLI resource management system. You can also specify an OBS path, for example, **obs://***Bucket name***/***Package name*.

Parameter	Mandatory	Type	Description
className	Yes	String	Java/Spark main class of the batch processing job.
queue	No	String	Queue name. Set this parameter to the name of the created DLI queue. The queue must be of the general-purpose type. **NOTE** • This parameter is compatible with the **cluster_name** parameter. That is, if **cluster_name** is used to specify a queue, the queue is still valid. • You are advised to use the **queue** parameter. The **queue** and **cluster_name** parameters cannot coexist.
cluster_name	No	String	Queue name. Set this parameter to the created DLI queue name. **NOTE** You are advised to use the **queue** parameter. The **queue** and **cluster_name** parameters cannot coexist.
args	No	Array of Strings	Input parameters of the main class, that is, application parameters.
sc_type	No	String	Compute resource type. Currently, resource types A, B, and C are available. If this parameter is not specified, the minimum configuration (type A) is used. For details about resource types, see **Table 14-3**.
jars	No	Array of Strings	Name of the package that is of the JAR type and has been uploaded to the DLI resource management system. You can also specify an OBS path, for example, **obs://***Bucket name***/***Package name*.
pyFiles	No	Array of Strings	Name of the package that is of the PyFile type and has been uploaded to the DLI resource management system. You can also specify an OBS path, for example, **obs://***Bucket name***/***Package name*.
files	No	Array of Strings	Name of the package that is of the file type and has been uploaded to the DLI resource management system. You can also specify an OBS path, for example, **obs://***Bucket name***/***Package name*.

Parameter	Mandatory	Type	Description
modules	No	Array of Strings	Name of the dependent system resource module. You can view the module name using the API related to **Querying Resource Packages in a Group (Discarded)**. DLI provides dependencies for executing datasource jobs. The following table lists the dependency modules corresponding to different services. • CloudTable/MRS HBase: sys.datasource.hbase • CloudTable/MRS OpenTSDB: sys.datasource.opentsdb • RDS MySQL: sys.datasource.rds • RDS Postgre: preset • DWS: preset • CSS: sys.datasource.css
resources	No	Array of objects	JSON object list, including the name and type of the JSON package that has been uploaded to the queue. For details, see **Table 14-4**.
groups	No	Array of objects	JSON object list, including the package group resource. For details about the format, see the request example. If the type of the **name** in **resources** is not verified, the package with the name exists in the group. For details, see **Table 14-5**.
conf	No	Object	**Batch** configuration item. For details, see **Spark Configuration**.
name	No	String	Batch processing task name. The value contains a maximum of 128 characters.
driverMemory	No	String	Driver memory of the Spark application, for example, **2 GB** and **2048 MB**. This configuration item replaces the default parameter in **sc_type**. The unit must be provided. Otherwise, the startup fails.
driverCores	No	Integer	Number of CPU cores of the Spark application driver. This configuration item replaces the default parameter in **sc_type**.

Parameter	Mandatory	Type	Description
executorMemory	No	String	Executor memory of the Spark application, for example, **2 GB** and **2048 MB**. This configuration item replaces the default parameter in **sc_type**. The unit must be provided. Otherwise, the startup fails.
executorCores	No	Integer	Number of CPU cores of each Executor in the Spark application. This configuration item replaces the default parameter in **sc_type**.
numExecutors	No	Integer	Number of Executors in a Spark application. This configuration item replaces the default parameter in **sc_type**.
obs_bucket	No	String	OBS bucket for storing the Spark jobs. Set this parameter when you need to save jobs.
auto_recovery	No	Boolean	Whether to enable the retry function. If enabled, Spark jobs will be automatically retried after an exception occurs. The default value is **false**.
max_retry_times	No	Integer	Maximum retry times. The maximum value is **100**, and the default value is **20**.
feature	No	String	Job feature. Type of the Spark image used by a job. • **basic**: indicates that the basic Spark image provided by DLI is used. • **custom**: indicates that the user-defined Spark image is used. • **ai**: indicates that the AI image provided by DLI is used.
spark_version	No	String	Version of the Spark component • If the in-use Spark version is 2.3.2, this parameter is not required. • If the current Spark version is 2.3.3, this parameter is required when **feature** is **basic** or **ai**. If this parameter is not set, the default Spark version 2.3.2 is used.

Parameter	Mandatory	Type	Description
image	No	String	Custom image. The format is **Organization name/Image name:Image version**. This parameter is valid only when **feature** is set to **custom**. You can use this parameter with the **feature** parameter to specify a user-defined Spark image for job running. For details about how to use custom images, see .
catalog_name	No	String	To access metadata, set this parameter to **dli**.

Table 14-3 Resource types

Resource Type	Physical Resource	driverCores	executorCores	driverMemory	executorMemory	numExecutor
A	8 vCPUs, 32-GB memory	2	1	7 GB	4 GB	6
B	16 vCPUs, 64-GB memory	2	2	7 GB	8 GB	7
C	32 vCPUs, 128-GB memory	4	2	15 GB	8 GB	14

Table 14-4 resources parameters

Parameter	Mandatory	Type	Description
name	No	String	Resource name You can also specify an OBS path, for example, **obs://***Bucket name***/***Package name*.
type	No	String	Resource type.

Table 14-5 groups parameters

Parameter	Mandatory	Type	Description
name	No	String	User group name
resources	No	Array of objects	User group resource For details, see **Table 14-4**.

Response

Table 14-6 Response parameters

Parameter	Mandatory	Type	Description
id	No	String	ID of a batch processing job.
appId	No	String	Back-end application ID of a batch processing job.
name	No	String	Batch processing task name. The value contains a maximum of 128 characters.
owner	No	String	Owner of a batch processing job.
proxyUser	No	String	Proxy user (resource tenant) to which a batch processing job belongs.
state	No	String	Status of a batch processing job. For details, see **Table 14-7**.
kind	No	String	Type of a batch processing job. Only Spark parameters are supported.
log	No	Array of strings	Last 10 records of the current batch processing job.
sc_type	No	String	Type of a computing resource. If the computing resource type is customized, value **CUSTOMIZED** is returned.
cluster_name	No	String	Queue where a batch processing job is located.

Parameter	Mandatory	Type	Description
queue	Yes	String	Queue name. Set this parameter to the name of the created DLI queue. **NOTE** • This parameter is compatible with the **cluster_name** parameter. That is, if **cluster_name** is used to specify a queue, the queue is still valid. • You are advised to use the **queue** parameter. The **queue** and **cluster_name** parameters cannot coexist.
image	No	String	Custom image. The format is **Organization name/Image name:Image version**. This parameter is valid only when **feature** is set to **custom**. You can use this parameter with the **feature** parameter to specify a user-defined Spark image for job running. For details about how to use custom images, see .
create_time	No	Long	Time when a batch processing job is created. The timestamp is expressed in milliseconds.
update_time	No	Long	Time when a batch processing job is updated. The timestamp is expressed in milliseconds.
duration	No	Long	Job running duration (unit: millisecond)

Table 14-7 Batch processing job statuses

Parameter	Type	Description
starting	String	The batch processing job is being started.
running	String	The batch processing job is executing a task.
dead	String	The batch processing job has exited.
success	String	The batch processing job is successfully executed.
recovering	String	The batch processing job is being restored.

Example Request

Create a Spark job. Set the Spark main class of the job to **org.apache.spark.examples.SparkPi**, specify the program package to **batchTest/**

spark-examples_2.11-2.1.0.luxor.jar, and load the program package whose type is **jar** and the resource package whose type is **files**.

```
{
    "file": "batchTest/spark-examples_2.11-2.1.0.luxor.jar",
    "className": "org.apache.spark.examples.SparkPi",
    "sc_type": "A",
    "jars": ["demo-1.0.0.jar"],
    "files": ["count.txt"],
    "resources":[
                {"name": "groupTest/testJar.jar", "type": "jar"},
                {"name": "kafka-clients-0.10.0.0.jar", "type": "jar"}],
    "groups": [
                {"name": "groupTestJar", "resources": [{"name": "testJar.jar", "type": "jar"}, {"name":
"testJar1.jar", "type": "jar"}]},
                {"name": "batchTest", "resources":  [{"name": "luxor.jar", "type": "jar"}]}],
    "queue": " test",
    "name": "TestDemo4",
    "feature": "basic",
    "spark_version": "2.3.2"
}
```

NOTE

The **batchTest/spark-examples_2.11-2.1.0.luxor.jar** file has been uploaded through API involved in **Uploading a Package Group (Discarded)**.

Example Response

```
{
  "id": "07a3e4e6-9a28-4e92-8d3f-9c538621a166",
  "appId": "",
  "name": "",
  "owner": "test1",
  "proxyUser": "",
  "state": "starting",
  "kind": "",
  "log": [],
  "sc_type": "CUSTOMIZED",
  "cluster_name": "aaa",
  "queue": "aaa",
  "create_time": 1607589874156,
  "update_time": 1607589874156
}
```

Status Codes

Table 14-8 describes the status code.

Table 14-8 Status code

Status Code	Description
200	The job is created successfully.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

14.2 Listing Batch Processing Jobs

Function

This API is used to list batch processing jobs in a queue of a project.

URI

- URI format

  GET /v2.0/{project_id}/batches

- Parameter description

Table 14-9 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 14-10 query parameter description

Parameter	Mandatory	Type	Description
job_name	No	String	Name of a batch processing job.
job-id	No	String	This API is used to query a batch job based on the job ID.
cluster_name	No	String	DLI queue name. If this parameter is left blank, the names of all batch processing jobs in the current project are obtained. You are advised to specify this parameter, instead of leaving it blank.
queue_name	No	String	DLI queue name. You can query batch jobs based on the queue name. This method is recommended.
from	No	Integer	Index number of the start batch processing job. By default, the index number starts from 0.

Parameter	Mandatory	Type	Description
size	No	Integer	Number of batch processing jobs to be queried
state	No	String	Query batch jobs by job status.
owner	No	String	User who submits a job.

Request

None

Response

Table 14-11 Response parameters

Parameter	Mandatory	Type	Description
from	No	Integer	Index number of the start batch processing job.
total	No	Integer	Total number of batch processing jobs.
sessions	No	Array of objects	Batch job information. For details, see **Table 14-12**.
create_time	No	Long	Time when a batch processing job is created.

Table 14-12 sessions parameters

Parameter	Mandatory	Type	Description
duration	No	Long	Job running duration (unit: millisecond)
id	No	String	ID of a batch processing job.
state	No	String	Status of a batch processing job
appId	No	String	Back-end application ID of a batch processing job
log	No	Array of Strings	Last 10 records of the current batch processing job
sc_type	No	String	Type of a computing resource. If the computing resource type is customized, value **CUSTOMIZED** is returned.

Parameter	Mandatory	Type	Description
cluster_name	No	String	Queue where a batch processing job is located.
create_time	No	Long	Time when a batch processing job is created. The timestamp is in milliseconds.
name	No	String	Name of a batch processing job.
owner	No	String	Owner of a batch processing job.
proxyUser	No	String	Proxy user (resource tenant) to which a batch processing job belongs.
kind	No	String	Type of a batch processing job. Only Spark parameters are supported.
queue	No	String	Queue where a batch processing job is located.
image	No	String	Custom image. The format is **Organization name/Image name:Image version**. This parameter is valid only when **feature** is set to **custom**. You can use this parameter with the **feature** parameter to specify a user-defined Spark image for job running.
req_body	No	String	Request parameter details.
update_time	No	Long	Time when a batch processing job is updated. The timestamp is in milliseconds.

Example Request

None

Example Response

```
{
  "from": 0,
  "total": 1,
  "sessions": [
    {
      "id": "178fa687-2e8a-41ed-a439-b00de60bb176",
      "state": "dead",
      "appId": null,
      "log": [
       "stdout: ",
       "stderr: ",
       "YARN Diagnostics: "
      ],
      "sc_type": "A",
      "cluster_name": "test",
      "create_time": 1531906043036
    }
```



```
  ]
}
```

Status Codes

Table 14-13 describes the status code.

Table 14-13 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

14.3 Querying Batch Job Details

Function

This API is used to query details about a batch processing job based on the job ID.

URI

- URI format

  GET /v2.0/{project_id}/batches/{batch_id}
- Parameter description

  Table 14-14 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
batch_id	Yes	String	ID of a batch processing job.

Request

None

Response

Table 14-15 Response parameters

Parameter	Mandatory	Type	Description
id	No	String	ID of a batch processing job.
appId	No	String	Back-end application ID of a batch processing job.
name	No	String	Name of a batch processing job.
owner	No	String	Owner of a batch processing job.
proxyUser	No	String	Proxy user (resource tenant) to which a batch processing job belongs.
state	No	String	Status of a batch processing job. For details, see **Table 14-7** in **Creating a Batch Processing Job**.
kind	No	String	Type of a batch processing job. Only Spark parameters are supported.
log	No	Array of Strings	Last 10 records of the current batch processing job.
sc_type	No	String	Type of a computing resource. If the computing resource type is customized, value **CUSTOMIZED** is returned.
cluster_name	No	String	Queue where a batch processing job is located.
queue	No	String	Queue where a batch processing job is located.
create_time	No	Long	Time when a batch processing job is created. The timestamp is expressed in milliseconds.
update_time	No	Long	Time when a batch processing job is updated. The timestamp is expressed in milliseconds.
feature	No	String	Job feature. Type of the Spark image used by a job. • **basic**: indicates that the basic Spark image provided by DLI is used. • **custom**: indicates that the user-defined Spark image is used. • **ai**: indicates that the AI image provided by DLI is used.

Parameter	Mandatory	Type	Description
spark_version	No	String	Version of the Spark component used by a job. Set this parameter when **feature** is set to **basic** or **ai**. If this parameter is not set, the default Spark component version 2.3.2 is used.
image	No	String	Custom image. The format is **Organization name/Image name:Image version**. This parameter is valid only when **feature** is set to **custom**. You can use this parameter with the **feature** parameter to specify a user-defined Spark image for job running. For details about how to use custom images, see .
req_body	No	String	Request parameter details.

Example Request

None

Example Response

```
{
   "id": "0a324461-d9d9-45da-a52a-3b3c7a3d809e",
   "appId": "",
   "name": "",
   "owner": "",
   "proxyUser": "",
   "state": "starting",
   "kind":"",
   "log": [
         "stdout: ",
          "stderr: ",
          "YARN Diagnostics: "
   ],
   "sc_type": "A",
   "cluster_name": "test",
   "queue": "test",
   "create_time": 1531906043036,
   "update_time": 1531906043036
}
```

Status Codes

Table 14-16 describes the status code.

Table 14-16 Status codes

Status Code	Description
200	The query is successful.

Status Code	Description
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

14.4 Querying a Batch Job Status

Function

This API is used to obtain the execution status of a batch processing job.

URI

- URI format

  GET /v2.0/{project_id}/batches/{batch_id}/state
- Parameter description

  Table 14-17 URI parameters

Parameter	Mandatory	Description
project_id	Yes	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
batch_id	Yes	ID of a batch processing job.

Request

None

Response

Table 14-18 Response parameters

Parameter	Mandatory	Type	Description
id	No	String	ID of a batch processing job, which is in the universal unique identifier (UUID) format.
state	No	String	Status of a batch processing job. For details, see **Table 14-7** in **Creating a Batch Processing Job**.

Example Request

None

Example Response

```
{"id":"0a324461-d9d9-45da-a52a-3b3c7a3d809e","state":"Success"}
```

Status Codes

Table 14-19 describes the status code.

Table 14-19 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

14.5 Canceling a Batch Processing Job

Function

This API is used to cancel a batch processing job.

NOTE

Batch processing jobs in the **Successful** or **Failed** state cannot be canceled.

URI

- URI format

  DELETE /v2.0/{project_id}/batches/{batch_id}

- Parameter description

Table 14-20 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
batch_id	Yes	String	ID of a batch processing job. Set the value to the job ID obtained in **Creating a Batch Processing Job**.

Request

None

Response

Table 14-21 Response parameter

Parameter	Mandatory	Type	Description
msg	No	String	If the batch processing job is successfully canceled, value **deleted** is returned.

Example Request

None

Example Response

```
{
    "msg": "deleted"
}
```


Status Codes

Table 14-22 describes the status code.

Table 14-22 Status codes

Status Code	Description
200	Deletion succeeded.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

15 APIs Related to Spark Job Templates

15.1 Creating a Job Template

Function

This API is used to create a job template.

URI

- URI format
  POST /v3/{project_id}/templates
- Parameter description

Table 15-1 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 15-2 Parameters

Parameter	Mandatory	Type	Description
type	Yes	String	Template type. Available values are as follows: • **SQL**: SQL template • **SPARK**: Spark template
name	Yes	String	Template name
body	Yes	String	Template content
group	No	String	Template group name
description	No	String	Description of the template
language	No	String	Language

Response

Table 15-3 Parameters

Parameter	Type	Description
id	String	Template ID

Example Request

Create a SQL job template whose content is **test sql**.

```
{
   "type": "SQL",
   "name": "test",
   "description": "",
   "body": "test aql",
   "group": ""
}
```

Example Response

```
{
   "id":"85aad9e5-3766-46c2-b3c0-16c2832e52e8"
}
```


Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

15.2 Listing Job Templates

Function

This API is used to list job templates.

URI

- URI format

  GET /v3/{project_id}/templates?type=spark
- Parameter description

Table 15-4 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 15-5 query parameters

Parameter	Mandatory	Type	Description
type	Yes	String	Template type. Available values: • **SPARK**: Spark template Currently, only Spark templates are supported.
keyword	No	String	Keyword used for searching for template names. Fuzzy match is supported.

Parameter	Mandatory	Type	Description
page-size	No	Integer	Maximum number of lines displayed on each page. Value range: [1, 100]. The default value is **50**.
current-page	No	Integer	Current page number. The default value is **1**.

Request

None

Response

Table 15-6 Parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	String	System prompt. If execution succeeds, the message may be left blank.
count	Integer	Number of returned templates
templates	Array of objects	Template information list. For details, see **Table 15-7**.

Table 15-7 templates parameters

Parameter	Type	Description
type	String	Template type
id	String	Template ID
name	String	Template name
body	Object	Template content. For details, see **Table 15-8**.
group	String	Template group
description	String	Description of the template
language	String	Language

Parameter	Type	Description
owner	String	Creator of the template

Table 15-8 body parameters

Parameter	Type	Description
file	String	Name of the package that is of the JAR or pyFile type and has been uploaded to the DLI resource management system. You can also specify an OBS path, for example, **obs://***Bucket name***/***Package name*.
className	String	Java/Spark main class of the template.
cluster_name	String	Queue name. Set this parameter to the created DLI queue name. **NOTE** You are advised to use the **queue** parameter. The **queue** and **cluster_name** parameters cannot coexist.
args	Array of Strings	Input parameters of the main class, that is, application parameters.
sc_type	String	Compute resource type. Currently, resource types A, B, and C are available. If this parameter is not specified, the minimum configuration (type A) is used. For details about resource types, see **Table 14-3**.
jars	Array of Strings	Name of the package that is of the JAR type and has been uploaded to the DLI resource management system. You can also specify an OBS path, for example, **obs://***Bucket name***/***Package name*.
pyFiles	Array of Strings	Name of the package that is of the PyFile type and has been uploaded to the DLI resource management system. You can also specify an OBS path, for example, **obs://***Bucket name***/***Package name*.

Parameter	Type	Description
files	Array of Strings	Name of the package that is of the file type and has been uploaded to the DLI resource management system. You can also specify an OBS path, for example, **obs://***Bucket name***/***Package name*.
modules	Array of Strings	Name of the dependent system resource module. You can view the module name using the API related to **Querying Resource Packages in a Group (Discarded)**. DLI provides dependencies for executing datasource jobs. The following table lists the dependency modules corresponding to different services. • CloudTable/MRS HBase: sys.datasource.hbase • CloudTable/MRS OpenTSDB: sys.datasource.opentsdb • RDS MySQL: sys.datasource.rds • RDS Postgre: preset • DWS: preset • CSS: sys.datasource.css
resources	Array of objects	JSON object list, including the name and type of the JSON package that has been uploaded to the queue. For details, see **Table 14-4**.
groups	Array of objects	JSON object list, including the package group resource. For details about the format, see the request example. If the type of the **name** in **resources** is not verified, the package with the name exists in the group. For details, see **Table 14-5**.
conf	Object	**Batch** configuration item. For details, see **Spark Configuration**.
name	String	Batch processing task name. The value contains a maximum of 128 characters.

Parameter	Type	Description
driverMemory	String	Driver memory of the Spark application, for example, **2 GB** and **2048 MB**. This configuration item replaces the default parameter in **sc_type**. The unit must be provided. Otherwise, the startup fails.
driverCores	Integer	Number of CPU cores of the Spark application driver. This configuration item replaces the default parameter in **sc_type**.
executorMemory	String	Executor memory of the Spark application, for example, **2 GB** and **2048 MB**. This configuration item replaces the default parameter in **sc_type**. The unit must be provided. Otherwise, the startup fails.
executorCores	Integer	Number of CPU cores of each Executor in the Spark application. This configuration item replaces the default parameter in **sc_type**.
numExecutors	Integer	Number of Executors in a Spark application. This configuration item replaces the default parameter in **sc_type**.
obs_bucket	String	OBS bucket for storing the Spark jobs. Set this parameter when you need to save jobs.
auto_recovery	Boolean	Whether to enable the retry function. If enabled, Spark jobs will be automatically retried after an exception occurs. The default value is **false**.
max_retry_times	Integer	Maximum retry times. The maximum value is **100**, and the default value is **20**.
feature	String	Job feature. Type of the Spark image used by a job. • **basic**: indicates that the basic Spark image provided by DLI is used. • **custom**: indicates that the user-defined Spark image is used. • **ai**: indicates that the AI image provided by DLI is used.

Parameter	Type	Description
spark_version	String	Version of the Spark component • If the in-use Spark version is 2.3.2, this parameter is not required. • If the current Spark version is 2.3.3, this parameter is required when **feature** is **basic** or **ai**. If this parameter is not set, the default Spark version 2.3.2 is used.
image	String	Custom image. The format is **Organization name/Image name:Image version**.
queue	String	Queue name. Set this parameter to the name of the created DLI queue. The queue must be of the general-purpose type. **NOTE** • This parameter is compatible with the **cluster_name** parameter. That is, if **cluster_name** is used to specify a queue, the queue is still valid. • You are advised to use the **queue** parameter. The **queue** and **cluster_name** parameters cannot coexist.
catalog_name	String	To access metadata, set this parameter to **dli**.

Example Request

None

Example Response

```
{
    "is_success": true,
    "message": "",
    "templates": [
        {
            "name": "test2",
            "body": {
                "auto_recovery": false,
                "max_retry_times": 20,
            },
            "group": "",
            "description": "",
            "type": "SPARK",
            "id": "3c92c202-b17c-4ed7-b353-ea08629dd671"
        }
    ],
    "count": 1
}
```


Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

15.3 Modifying a Job Template

Function

This API is used to modify a job template.

URI

- URI format
  PUT /v3/{project_id}/templates/{template_id}
- Parameter description

Table 15-9 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
template_id	Yes	String	Template ID.

Request

Table 15-10 Parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Template name
body	Yes	String	Template content
group	No	String	Template group
description	No	String	Description of the template

Response

Table 15-11 Parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	String	System prompt. If execution succeeds, the message may be left blank.

Example Request

Modify the name and content of a job template.

```
{
   "name": "test1",
   "body":"ss"
 }
```

Example Response

```
{
  "is_success": true,
  "message": ""
 }
```

Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

15.4 Obtaining a Job Template

Function

This API is used to obtain a job template.

URI

- URI format
  GET /v3/{project_id}/templates/{template_id}
- Parameter description

Table 15-12 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
template_id	Yes	String	Template ID.

Request

None

Response

Table 15-13 Parameters

Parameter	Type	Description
type	String	Template type
id	String	Template ID
name	String	Template name
body	Object	Template content For details, see **Table 15-8**.
group	String	Template group
description	String	Description of the template
language	String	Language
owner	String	Creator of the template

Example Request

None

Example Response

```
{
  "name": "test2",
```



```
    "body": {
        "auto_recovery": false,
        "max_retry_times": 20,
    },
    "group": "",
    "description": "",
    "type": "SPARK",
    "id": "3c92c202-b17c-4ed7-b353-ea08629dd671"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

16 Permissions Policies and Supported Actions

This section describes fine-grained permissions management for your DLI. If your Huawei Cloud account does not need individual IAM users, then you may skip this section.

By default, new IAM users do not have permissions assigned. You need to add them to one or more groups, and attach permissions policies or roles to these groups. Users inherit permissions from the groups to which they are added After authorization, the user can perform specified operations on MRS based on the permissions.

You can grant users permissions by using roles and policies.

- Roles are a type of coarse-grained authorization mechanism that defines permissions related to user responsibilities.
- Policies define API-based permissions for operations on specific resources under certain conditions, allowing for more fine-grained, secure access control of cloud resources.

NOTE

Policy-based authorization is useful if you want to allow or deny the access to an API.

An account has all the permissions required to call all APIs, but IAM users must be assigned the required permissions. The permissions required for calling an API are determined by the actions supported by the API. Only users who have been granted permissions allowing the actions can call the API successfully. For example, if an IAM user needs to create buckets using an API, the user must have been granted permissions that allow the **dli:queue:create_queue** action.

Supported Actions

VBS provides system-defined policies that can be directly used in IAM. You can also create custom policies and use them to supplement system-defined policies, implementing more refined access control. Operations supported by policies are specific to APIs. The following are common concepts related to policies:

- Permissions: Allow or deny operations on specified resources under specific conditions.

- APIs: RESTful APIs that can be called in a custom policy.
- Actions: added to a custom policy to control permissions for specific operations.
- Related actions: Actions on which a specific action depends to take effect. When assigning permissions for the action to a user, you also need to assign permissions for the related actions.
- IAM or enterprise projects: Type of projects for which an action will take effect. Policies that contain actions supporting both IAM and enterprise projects can be assigned to user groups and take effect in both IAM and Enterprise Management. Policies that only contain actions supporting IAM projects can be assigned to user groups and only take effect for IAM. Such policies will not take effect if they are assigned to user groups in Enterprise Project.

NOTE

The check mark (√) indicates that an action takes effect. The cross mark (x) indicates that an action does not take effect.

DLI supports the following actions that can be defined in custom policies:

Table 16-1 Actions

Permission	API	Actions	IAM Project (Project)	Enterprise Project (Enterprise Project)
Creating a queue	POST /v1.0/{project_id}/queues	dli:queue:createQueue	√	√
Deleting a queue	DELETE /v1.0/{project_id}/queues/{queue_name}	dli:queue:dropQueue	√	√
Submitting a job	POST /v1.0/{project_id}/jobs/submit-job	dli:queue:submitJob	√	×
Canceling a job	DELETE /v1.0/{project_id}/jobs/{job_id}	dli:queue:cancelJob	√	×
Viewing queue permissions of other users	GET /v1.0/{project_id}/queues/{queue_name}/users	dli:queue:showPrivileges	√	×
Restarting a queue	PUT /v1.0/{project_id}/queues/{queue_name}/action	dli:queue:restart	√	×

Permission	API	Actions	IAM Project (Project)	Enterprise Project (Enterprise Project)
Scaling out/in a queue	PUT /v1.0/{project_id}/queues/{queue_name}/action	dli:queue:scaleQueue	√	×
Granting elastic resource pool permissions	PUT /v1.0/{project_id}/user-authorization	dli:elasticresourcepool:grantPrivilege	√	×
Revoking elastic resource pool permissions	PUT /v1.0/{project_id}/user-authorization	dli:elasticresourcepool:revokePrivilege	√	×
Creating a database	POST /v1.0/{project_id}/databases	dli:database:createDatabase	√	×
Deleting a database	DELETE /v1.0/{project_id}/databases/{database_name}	dli:database:dropDatabase	√	×
Modifying database configuration	POST /v1.0/{project_id}/jobs/submit-job	dli:database:alterDatabaseProperties	√	×
Explaining the SQL statement as an execution plan	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:database:explain	√	×
Creating a table	POST /v1.0/{project_id}/databases/{database_name}/tables	dli:database:createTable	√	×
Creating a view	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:database:createView	√	×

Permission	API	Actions	IAM Project (Project)	Enterprise Project (Enterprise Project)
Creating a function	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:database:createFunction	√	×
Describing a function	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:database:describeFunction	√	×
Deleting a function	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:database:dropFunction	√	×
Displaying a function	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:database:showFunctions	√	×
Creating a role	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:database:createRole	√	×
Deleting a role	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:database:dropRole	√	×
Displaying a role	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:database:showRoles	√	×
Displaying all roles	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:database:showAllRoles	√	×
Binding a role	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:database:grantRole	√	×
Unbinding a role	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:database:revokeRole	√	×

Permission	API	Actions	IAM Project (Project)	Enterprise Project (Enterprise Project)
Displaying the binding relationships between all roles and users	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:database:showUsers	√	×
Viewing database permissions of other users	GET /v1.0/{project_id}/databases/{database_name}/users	dli:database:showPrivileges	√	×
Displaying a database	GET /v1.0/{project_id}/databases	dli:database:displayDatabase	√	×
Displaying all databases	GET /v1.0/{project_id}/databases	dli:database:displayAllDatabases	√	×
Displaying all tables	GET /v1.0/{project_id}/databases	dli:database:displayAllTables	√	×
Granting database permissions to a specified user	PUT /v1.0/{project_id}/user-authorization	dli:database:grantPrivilege	√	×
Removing database permissions of a specified user	PUT /v1.0/{project_id}/user-authorization	dli:database:revokePrivilege	√	×
Deleting a table	DELETE /v1.0/{project_id}/databases/{database_name}/tables/{table_name}	dli:table:dropTable	√	×

Permission	API	Actions	IAM Project (Project)	Enterprise Project (Enterprise Project)
Displaying the table structure	GET /v1.0/{project_id}/databases/{database_name}/tables/{table_name}	dli:table:describeTable	√	×
Querying a table	GET /v1.0/{project_id}/databases/{database_name}/tables/{table_name}/preview	dli:table:select	√	×
Displaying table configuration	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:table:showTableProperties	√	×
Displaying the table creation statement	GET /v1.0/{project_id}/databases/{database_name}/tables/{table_name}/preview	dli:table:showCreateTable	√	×
Displaying all partitions	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:table:showPartitions	√	×
Setting table configuration	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:table:alterTableSetProperties	√	×
Adding a column	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:table:alterTableAddColumns	√	×
Adding partitions to a partitioned table	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:table:alterTableAddPartition	√	×
Renaming a table partition	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:table:alterTableRenamePartition	√	×

Permission	API	Actions	IAM Project (Project)	Enterprise Project (Enterprise Project)
Deleting partitions from a partitioned table	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:table:alterTableDropPartition	√	×
Restoring a table partition	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:table:alterTableRecoverPartition	√	×
Renaming a table	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:table:alterTableRename	√	×
Setting the partition path	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:table:alterTableSetLocation	√	×
Inserting table data	POST /v1.0/{project_id}/jobs/submit-job, statement invoking	dli:table:insertIntoTable	√	×
Rewriting table data	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:table:insertOverwriteTable	√	×
Viewing table permissions of other users	GET /v1.0/{project_id}/databases/{database_name}/tables/{table_name}/users	dli:table:showPrivileges	√	×
Clearing a table	POST /v1.0/{project_id}/jobs/submit-job	dli:table:truncateTable	√	×
Updating a table	POST /v1.0/{project_id}/jobs/submit-job	dli:table:update	√	×
Deleting table data	POST /v1.0/{project_id}/jobs/submit-job	dli:table:delete	√	×

Permission	API	Actions	IAM Project (Project)	Enterprise Project (Enterprise Project)
Modifying column information	POST /v1.0/{project_id}/jobs/submit-job	dli:table:alterTableChangeColumn	√	×
Deleting a column	POST /v1.0/{project_id}/jobs/submit-job	dli:table:alterTableDropColumns	√	×
Displaying data segments	POST /v1.0/{project_id}/jobs/submit-job	dli:table:showSegments	√	×
Merging data segments	POST /v1.0/{project_id}/jobs/submit-job	dli:table:compaction	√	×
Modifying a view	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:table:alterView	√	×
Displaying the table structure	POST /v1.0/{project_id}/jobs/submit-job, executing corresponding SQL statements	dli:table:describeTable	√	×
Granting data table permissions to a specified user	PUT /v1.0/{project_id}/user-authorization	dli:table:grantPrivilege	√	×
Removing data table permissions of a specified user	PUT /v1.0/{project_id}/user-authorization	dli:table:revokePrivilege	√	×

Permission	API	Actions	IAM Project (Project)	Enterprise Project (Enterprise Project)
Viewing the security authentication information permission list of other users	GET /v1.0/{project_id}/datasource/auth-infos/{auth_name}/users	dli:datasourceauth:showPrivileges	√	×
Using security authentication information	POST /v1.0/{project_id}/jobs/submit-job	dli:datasourceauth:useAuth	√	×
Deleting security authentication information	DELETE /v2.0/{project_id}/datasource/auth-infos/{auth_info_name}	dli:datasourceauth:dropAuth	√	×
Updating security authentication information	PUT /v2.0/{project_id}/datasource/auth-infos	dli:datasourceauth:updateAuth	√	×
Granting security authentication permissions to a specified user	PUT /v1.0/{project_id}/user-authorization	dli:datasourceauth:grantPrivilege	√	×

Permission	API	Actions	IAM Project (Project)	Enterprise Project (Enterprise Project)
Removing security authentication permissions of a specified user	PUT /v1.0/{project_id}/user-authorization	dli:datasourceauth:revokePrivilege	√	×
Granting job permissions to a specified user	PUT /v1.0/{{project_id}}/authorization	dli:jobs:grantPrivilege	√	×
Removing job permissions of a specified user	PUT /v1.0/{{project_id}}/authorization	dli:jobs:revokePrivilege	√	×
Querying a Column	POST /v1.0/{project_id}/jobs/submit-job, SQL statement invoking	dli:column:select	√	×
Granting permissions to a specified user queue	PUT /v1.0/{project_id}/user-authorization	dli:column:grantPrivilege	√	×
Removing permissions of a specified user queue	PUT /v1.0/{project_id}/user-authorization	dli:column:revokePrivilege	√	×
Listing Flink jobs	GET /v1.0/{project_id}/streaming/jobs	dli:jobs:listAll	√	×

Permission	API	Actions	IAM Project (Project)	Enterprise Project (Enterprise Project)
Querying Flink job details	GET /v1.0/{project_id}/streaming/jobs/{job_id}	dli:jobs:get	√	×
Creating a Flink SQL job	POST /v1.0/{project_id}/streaming/sql-jobs	dli:jobs:create	√	×
Creating a Flink Jar job	POST /v1.0/{project_id}/streaming/flink-jobs	dli:jobs:create	√	×
Updating a Flink SQL job	PUT /v1.0/{project_id}/streaming/sql-jobs/{job_id}	dli:jobs:update	√	×
Updating a Flink Jar job	PUT /v1.0/{project_id}/streaming/flink-jobs/{job_id}	dli:jobs:update	√	×
Deleting a Flink job	DELETE /v1.0/{project_id}/streaming/jobs/{job_id}	dli:jobs:delete	√	×
Running Flink jobs in batches	POST /v1.0/{project_id}/streaming/jobs/run	dli:jobs:start	√	×
Stopping a Flink job	POST /v1.0/{project_id}/streaming/jobs/stopob_id/stop	dli:jobs:stop	√	×

17 Out-of-Date APIs

17.1 Agency-related APIs (Discarded)

17.1.1 Obtaining DLI Agency Information (Discarded)

Function

This API is used to obtain the agency information of a DLI user.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  GET /v2/{project_id}/agency
- Parameter description

Table 17-1 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

None

Response

Table 17-2 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
version	No	String	Agency version information.
current_roles	No	Array of Strings	Role. The supported values are as follows: **obs_adm**: Administrator permissions for accessing and using the Object Storage Service. **dis_adm**: Administrator permissions for using Data Ingestion Service data as the data source **ctable_adm**: Administrator permissions for accessing and using the CloudTable service **vpc_netadm**: Administrator permissions for using the Virtual Private Cloud service **smn_adm**: Administrator permissions for using the Simple Message Notification service **te_admin**: Tenant Administrator permissions

Example Request

None

Example Response

```
{
    "is_success": true,
    "message": "",
    "version": "v2",
    "current_roles": [
        "ctable_adm",
        "vpc_netadm",
        "ief_adm",
        "dis_adm",
        "smn_adm",
        "obs_adm"
    ]
}
```

Status Codes

Table 17-3 describes status codes.

Table 17-3 Status codes

Status Code	Description
200	The agency information is obtained.
400	Request failure.
404	Not found.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 17-4 Error codes

Error Code	Error Message
DLI.0002	The object does not exist.
DLI.0999	An internal error occurre

17.1.2 Creating a DLI Agency (Discarded)

Function

This API is used to create an agency for a DLI user.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  POST /v2/{project_id}/agency
- Parameter description

Table 17-5 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-6 Request parameters

Parameter	Mandatory	Type	Description
roles	Yes	Array of Strings	Role. Currently, only **obs_adm**, **dis_adm**, **ctable_adm**, **vpc_netadm**, **smn_adm**, and **te_admin** are supported. • **obs_adm**: Administrator permissions for accessing and using the Object Storage Service. • **dis_adm**: Administrator permissions for using Data Ingestion Service data as the data source • **ctable_adm**: Administrator permissions for accessing and using the CloudTable service • **vpc_netadm**: Administrator permissions for using the Virtual Private Cloud service • **smn_adm**: Administrator permissions for using the Simple Message Notification service • **te_admin**: Tenant Administrator permissions

Response

Table 17-7 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.

Parameter	Mandatory	Type	Description
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Example Request

Create a DLI user agency. The agency has the following permissions: **CloudTable Administrator** for accessing and using CloudTable, **VPC Administrator** for using VPC, **DIS Administrator** for accessing and using DIS, **SMN Administrator** for using SMN, and accessing and using OBS.

```
{
    "roles": [
        "ctable_adm",
        "vpc_netadm",
        "dis_adm",
        "smn_adm",
        "obs_adm"
    ]
}
```

Example Response

```
{
    "is_success": true,
    "message": ""
}
```

Status Codes

Table 17-8 describes status codes.

Table 17-8 Status codes

Status Code	Description
200	The job is created successfully.
400	Request failure.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.2 Package Group-related APIs (Discarded)

17.2.1 Uploading a Package Group (Discarded)

Function

This API is used to upload a package group to a **project**. The function is similar to creating a package on the management console.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  POST /v2.0/{project_id}/resources

- Parameter description

Table 17-9 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-10 Request parameters

Parameter	Mandatory	Type	Description
paths	Yes	Array of Strings	List of OBS object paths. The OBS object path refers to the OBS object URL.
kind	Yes	String	File type of a package group. • **jar**: JAR file • **pyFile**: User Python file • **file**: User file • **modelFile**: User AI model file **NOTE** If the same group of packages to be uploaded contains different file types, select **file** as the type of the file to be uploaded.
group	Yes	String	Name of the group to be created.

Parameter	Mandatory	Type	Description
is_async	No	Boolean	Whether to upload resource packages in asynchronous mode. The default value is **false**, indicating that the asynchronous mode is not used. You are advised to upload resource packages in asynchronous mode.
tags	No	Array of objects	Resource tag. For details, see **Table 17-11**.

Table 17-11 tags parameter

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key. **NOTE** A tag key can contain a maximum of 128 characters. Only letters, digits, spaces, and special characters (_.:=+-@) are allowed, but the value cannot start or end with a space or start with **_sys_**.
value	Yes	String	**NOTE** A tag value can contain a maximum of 255 characters. Only letters, digits, spaces, and special characters (_.:=+-@) are allowed. The value cannot start or end with a space.

Response

Table 17-12 Response parameters

Parameter	Mandatory	Type	Description
group_name	No	String	Group name.
status	No	String	Status of a package group to be uploaded.
resources	No	Array of strings	List of names of resource packages contained in the group.
details	No	Array of body	Details about a group resource package. For details, see **Table 17-13**.
create_time	No	Long	UNIX timestamp when a package group is uploaded.

Parameter	Mandatory	Type	Description
update_time	No	Long	UNIX timestamp when a package group is updated.
is_async	No	Boolean	Whether to upload resource packages in asynchronous mode. The default value is **false**, indicating that the asynchronous mode is not used. You are advised to upload resource packages in asynchronous mode.
owner	No	String	Owner of a resource package.

Table 17-13 details parameter description

Parameter	Mandatory	Type	Description
create_time	Yes	Long	UNIX time when a resource package is uploaded. The timestamp is in milliseconds.
update_time	No	Long	UNIX time when the uploaded resource package is uploaded. The timestamp is in milliseconds.
resource_type	Yes	String	Resource type.
resource_name	No	String	Resource name.
status	No	String	• Value **UPLOADING** indicates that the resource package group is being uploaded. • Value **READY** indicates that the resource package has been uploaded. • Value **FAILED** indicates that the resource package fails to be uploaded.
underlying_name	No	String	Name of the resource packages in a queue.
is_async	No	Boolean	Whether to upload resource packages in asynchronous mode. The default value is **false**, indicating that the asynchronous mode is not used. You are advised to upload resource packages in asynchronous mode.

Example Request

Upload a JAR file from OBS to DLI and name the group **gatk**.


```
{
   "paths": [
      "https://xkftest.obs.xxx.com/txr_test/jars/spark-sdv-app.jar",
      "https://xkftest.obs.xxx.com/txr_test/jars/wordcount",
      "https://xkftest.obs.xxx.com/txr_test/jars/wordcount.py"
   ],
   "kind": "jar",
   "group": "gatk",
   "is_async":"true"
}
```

Example Response

```
{
   "group_name": "gatk",
   "status": "READY",
   "resources": [
      "spark-sdv-app.jar",
      "wordcount",
      "wordcount.py"
   ],
   "details": [
      {
         "create_time": 0,
         "update_time": 0,
         "resource_type": "jar",
         "resource_name": "spark-sdv-app.jar",
         "status": "READY",
         "underlying_name": "987e208d-d46e-4475-a8c0-a62f0275750b_spark-sdv-app.jar"
      },
      {
         "create_time": 0,
         "update_time": 0,
         "resource_type": "jar",
         "resource_name": "wordcount",
         "status": "READY",
         "underlying_name": "987e208d-d46e-4475-a8c0-a62f0275750b_wordcount"
      },
      {
         "create_time": 0,
         "update_time": 0,
         "resource_type": "jar",
         "resource_name": "wordcount.py",
         "status": "READY",
         "underlying_name": "987e208d-d46e-4475-a8c0-a62f0275750b_wordcount.py"
      }
   ],
   "create_time": 1551334579654,
   "update_time": 1551345369070
}
```

Status Codes

Table 17-14 describes the status code.

Table 17-14 Status codes

Status Code	Description
201	The file is successfully uploaded.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.2.2 Listing Package Groups (Discarded)

Function

This API is used to query all resources in a project, including groups.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  GET /v2.0/{project_id}/resources
- Parameter description

Table 17-15 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 17-16 query parameter description

Parameter	Mandatory	Type	Description
kind	No	String	Specifies the file type. The options are as follows: • **jar**: JAR file • **pyFile**: User Python file • **file**: User file • **modelFile**: User AI model file
tags	No	String	Specifies a label for filtering.

Request

None

Response

Table 17-17 Response parameters

Parameter	Mandatory	Type	Description
resources	No	Array of objects	List of names of uploaded user resources. For details about resources, see **Table 17-18**.
modules	No	Array of objects	List of built-in resource groups. For details about the groups, see **Table 17-19**.
groups	No	Array of objects	Uploaded package groups of a user.
total	Yes	Integer	Total number of returned resource packages.

Table 17-18 resources parameters

Parameter	Mandatory	Type	Description
create_time	No	Long	UNIX timestamp when a resource package is uploaded.
update_time	No	Long	UNIX timestamp when the uploaded resource package is uploaded.
resource_type	No	String	Resource type.
resource_name	No	String	Resource name.
status	No	String	• Value **UPLOADING** indicates that the resource package is being uploaded. • Value **READY** indicates that the resource package has been uploaded. • Value **FAILED** indicates that the resource package fails to be uploaded.
underlying_name	No	String	Name of the resource package in the queue.
owner	No	String	Owner of the resource package.

Table 17-19 modules parameters

Parameter	Mandatory	Type	Description
module_name	No	String	Module name.
module_type	No	String	Module type.
status	No	String	• Value **UPLOADING** indicates that the package group is being uploaded. • Value **READY** indicates that the package group has been uploaded. • Value **FAILED** indicates that the package group fails to be uploaded.
resources	No	Array of Strings	List of names of resource packages contained in the group.
description	No	String	Module description.
create_time	No	Long	UNIX timestamp when a package group is uploaded.
update_time	No	Long	UNIX timestamp when a package group is updated.

Example Request

None

Example Response

```
{
  "resources": [
    {
      "create_time": 1521532893736,
      "update_time": 1521552364503,
      "resource_type": "jar",
      "resource_name": "luxor-router-1.1.1.jar",
      "status": "READY",
      "underlying_name": "3efffb4f-40e9-455e-8b5a-a23b4d355e46_luxor-router-1.1.1.jar"
    }
  ],
  "groups": [
    {
      "group_name": "groupTest",
      "status": "READY",
      "resources": [
        "part-00000-9dfc17b1-2feb-45c5-b81d-bff533d6ed13.csv.gz",
        "person.csv"
      ],
      "details": [
        {
          "create_time": 1547090015132,
          "update_time": 1547090015132,
          "resource_type": "jar",
          "resource_name": "part-00000-9dfc17b1-2feb-45c5-b81d-bff533d6ed13.csv.gz",
          "status": "READY",
```



```
                "underlying_name": "db50c4dc-7187-4eb9-
a5d0-73ba8102ea5e_part-00000-9dfc17b1-2feb-45c5-b81d-bff533d6ed13.csv.gz"
            },
            {
                "create_time": 1547091098668,
                "update_time": 1547091098668,
                "resource_type": "file",
                "resource_name": "person.csv",
                "status": "READY",
                "underlying_name": "a4243a8c-bca6-4e77-a968-1f3b00217474_person.csv"
            }
        ],
        "create_time": 1547090015131,
        "update_time": 1547091098666
    }
  ],
  "modules": [
    {
        "module_name": "gatk",
        "status": "READY",
        "resources": [
            "gatk.jar",
            "tika-core-1.18.jar",
            "s3fs-2.2.2.jar"
        ],
        "create_time": 1521532893736,
        "update_time": 1521552364503
    }
  ]
}
```


Status Codes

Table 17-20 describes the status code.

Table 17-20 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.2.3 Uploading a JAR Package Group (Discarded)

Function

This API is used to upload a group of JAR packages to a **project**.

When a resource group with the same name is uploaded, the new group overwrites the old group.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  POST /v2.0/{project_id}/resources/jars

- Parameter description

Table 17-21 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-22 Request parameters

Parameter	Mandatory	Type	Description
paths	Yes	Array of Strings	List of OBS object paths. The OBS object path refers to the OBS object URL.
group	Yes	String	Name of a package group.

Response

Table 17-23 Response parameters

Parameter	Mandatory	Type	Description
group_name	No	String	Group name.
status	No	String	Status of a package group to be uploaded.
resources	No	Array of strings	List of names of resource packages contained in the group.
details	No	Array of body	Details about a group resource package. For details, see **Table 17-24**.

Parameter	Mandatory	Type	Description
create_time	No	Long	UNIX timestamp when a package group is uploaded.
update_time	No	Long	UNIX timestamp when a package group is updated.
is_async	No	Boolean	Whether to upload resource packages in asynchronous mode. The default value is **false**, indicating that the asynchronous mode is not used. You are advised to upload resource packages in asynchronous mode.
owner	No	String	Owner of a resource package.
description	No	String	Description of a resource module.
module_name	No	String	Name of a resource module.
module_type	No	String	Type of a resource module. • **jar**: User JAR file • **pyFile**: User Python file • **file**: User file

Table 17-24 details parameter description

Parameter	Mandatory	Type	Description
create_time	No	Long	UNIX time when a resource package is uploaded. The timestamp is in milliseconds.
update_time	No	Long	UNIX time when the uploaded resource package is uploaded. The timestamp is in milliseconds.
resource_type	No	String	Resource type. Set this parameter to **jar**.
resource_name	No	String	Resource name.

Parameter	Mandatory	Type	Description
status	No	String	• Value **UPLOADING** indicates that the resource package group is being uploaded. • Value **READY** indicates that the resource package has been uploaded. • Value **FAILED** indicates that the resource package fails to be uploaded.
underlying_name	No	String	Name of the resource packages in a queue.
is_async	No	Boolean	Indicates whether to upload a resource package asynchronously.

Example Request

Upload a JAR file from OBS to DLI and name the group **gatk**.

```
{
    "paths": [
        "https://test.obs.xxx.com/test_dli.jar"
    ],
    "group": "gatk"
}
```

Example Response

```
{
    "group_name": "gatk",
    "status": "READY",
    "resources": [
        "test_dli.jar"
    ],
    "details":[
        {
          "create_time":1608804435312,
          "update_time":1608804435312,
          "resource_type":"jar",
          "resource_name":"test_dli.jar",
          "status":"READY",
          "underlying_name":"test_dli.jar"
        }
       ],
    "create_time": 1521532893736,
    "update_time": 1521552364503,
    "is_async":false
}
```

Status Codes

Table 17-25 describes the status code.

Table 17-25 Status codes

Status Code	Description
201	Upload succeeded.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.2.4 Uploading a PyFile Package Group (Discarded)

Function

This API is used to upload a group of **PyFile** packages to a **project**. When a group with the same name as the PyFile package is uploaded, the new group overwrites the old group.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  POST /v2.0/{project_id}/resources/pyfiles
- Parameter description

Table 17-26 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-27 Request parameters

Parameter	Mandatory	Type	Description
paths	Yes	Array of strings	List of OBS object paths. The OBS object path refers to the OBS object URL.
group	Yes	String	Name of a package group.

Response

Table 17-28 Response parameters

Parameter	Mandatory	Type	Description
group_name	No	String	Group name.
status	No	String	Status of a package group to be uploaded.
resources	No	Array of strings	List of names of resource packages contained in the group.
details	No	Array of body	Details about a group resource package. For details, see **Table 17-29**.
create_time	No	Long	UNIX timestamp when a package group is uploaded.
update_time	No	Long	UNIX timestamp when a package group is updated.
is_async	No	Boolean	Whether to upload resource packages in asynchronous mode. The default value is **false**, indicating that the asynchronous mode is not used. You are advised to upload resource packages in asynchronous mode.
owner	No	String	Owner of a resource package.
description	No	String	Description of a resource module.
module_name	No	String	Name of a resource module.

Parameter	Mandatory	Type	Description
module_type	No	String	Type of a resource module. • **jar**: User JAR file • **pyFile**: User Python file • **file**: User file

Table 17-29 details parameter description

Parameter	Mandatory	Type	Description
create_time	No	Long	UNIX time when a resource package is uploaded. The timestamp is in milliseconds.
update_time	No	Long	UNIX time when the uploaded resource package is uploaded. The timestamp is in milliseconds.
resource_type	No	String	Resource type. Set this parameter to **pyFile**.
resource_name	No	String	Resource name.
status	No	String	• Value **UPLOADING** indicates that the resource package group is being uploaded. • Value **READY** indicates that the resource package has been uploaded. • Value **FAILED** indicates that the resource package fails to be uploaded.
underlying_name	No	String	Name of the resource packages in a queue.
is_async	No	Boolean	Indicates whether to upload a resource package asynchronously.

Example Request

Upload a Python file from OBS to DLI and name the group **gatk**.

```
{
    "paths": [
        "https://test.obs.xxx.com/dli_tf.py"
    ],
    "group": " gatk"
}
```


Example Response

```
{
   "group_name": "gatk",
   "status": "READY",
   "resources": [
      "dli_tf.py"
   ],
   "details":[
      {
        "create_time":1608804435312,
        "update_time":1608804435312,
        "resource_type":"pyFile",
        "resource_name":"dli_tf.py",
        "status":"READY",
        "underlying_name":"dli_tf.py"
      }
     ],
   "create_time": 1521532893736,
   "update_time": 1521552364503,
   "is_async":false
}
```

Status Codes

Table 17-30 describes the status code.

Table 17-30 Status codes

Status Code	Description
201	Upload succeeded.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.2.5 Uploading a File Package Group (Discarded)

Function

This API is used to upload a group of **File** packages to a **project**. When the **File** package group with the same name is uploaded, the new group overwrites the old group.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  POST /v2.0/{project_id}/resources/files
- Parameter description

  Table 17-31 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-32 Request parameters

Parameter	Mandatory	Type	Description
paths	Yes	Array of Strings	List of OBS object paths. The OBS object path refers to the OBS object URL.
group	Yes	String	Name of a package group.

Response

Table 17-33 Response parameters

Parameter	Mandatory	Type	Description
group_name	No	String	Group name.
status	No	String	Status of a package group to be uploaded.
resources	No	Array of strings	List of names of resource packages contained in the group.
details	No	Array of body	Details about a group resource package. For details, see **Table 17-34**.
create_time	No	Long	UNIX timestamp when a package group is uploaded.
update_time	No	Long	UNIX timestamp when a package group is updated.

Parameter	Mandatory	Type	Description
is_async	No	Boolean	Whether to upload resource packages in asynchronous mode. The default value is **false**, indicating that the asynchronous mode is not used. You are advised to upload resource packages in asynchronous mode.
owner	No	String	Owner of a resource package.
description	No	String	Description of a resource module.
module_name	No	String	Name of a resource module.
module_type	No	String	Type of a resource module. • **jar**: User JAR file • **pyFile**: User Python file • **file**: User file

Table 17-34 details parameter description

Parameter	Mandatory	Type	Description
create_time	No	Long	UNIX time when a resource package is uploaded. The timestamp is expressed in milliseconds.
update_time	No	Long	UNIX time when the uploaded resource package is uploaded. The timestamp is expressed in milliseconds.
resource_type	No	String	Resource type. Set this parameter to **file**.
resource_name	No	String	Resource name.
status	No	String	• Value **UPLOADING** indicates that the resource package group is being uploaded. • Value **READY** indicates that the resource package has been uploaded. • Value **FAILED** indicates that the resource package fails to be uploaded.
underlying_name	No	String	Name of the resource packages in a queue.

Parameter	Mandatory	Type	Description
is_async	No	Boolean	Indicates whether to upload a resource package asynchronously.

Example Request

Upload a file from OBS to DLI and name the group **gatk**.

```
{
    "paths": [
        "https: //test.obs.xxx.com/test_dli.jar",
        "https://test.obs.xxx.com/dli_tf.py"
    ],
    "group": "gatk"
}
```

Example Response

```
{
    "group_name": "gatk",
    "status": "READY",
    "resources": [
        "test_dli.jar",
        "dli_tf.py"
    ],
    "details":[
        {
          "create_time":1608804435312,
          "update_time":1608804435312,
          "resource_type":"file",
          "resource_name":"test_dli.jar",
          "status":"READY",
          "underlying_name":"test_dli.jar"
        },
       {
        "create_time":1608804435312,
        "update_time":1608804435312,
        "resource_type":"file",
        "resource_name":"dli_tf.py",
        "status":"READY",
        "underlying_name":"dli_tf.py"
      }
       ],
    "create_time": 1521532893736,
    "update_time": 1521552364503,
    "is_async":false
}
```

Status Codes

Table 17-35 describes the status code.

Table 17-35 Status codes

Status Code	Description
201	Upload succeeded.

Status Code	Description
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.2.6 Querying Resource Packages in a Group (Discarded)

Function

This API is used to query resource information of a package group in a **Project**.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  GET /v2.0/{project_id}/resources/{resource_name}
- Parameter description

Table 17-36 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
resource_name	Yes	String	Name of the resource package that is uploaded.

Table 17-37 query parameter description

Parameter	Mandatory	Type	Description
group	No	String	Name of the package group returned when the resource package is uploaded.

NOTE

The following is an example of the URL containing the **query** parameter:
GET /v2.0/{project_id}/resources/{resource_name}?group=*{group}*

Request

None

Response

Table 17-38 Response parameters

Parameter	Type	Description
create_time	Long	UNIX time when a resource package is uploaded. The timestamp is expressed in milliseconds.
update_time	Long	UNIX time when the uploaded resource package is uploaded. The timestamp is expressed in milliseconds.
resource_type	String	Resource type.
resource_name	String	Resource name.
status	String	• Value **UPLOADING** indicates that the resource package group is being uploaded. • Value **READY** indicates that the resource package has been uploaded. • Value **FAILED** indicates that the resource package fails to be uploaded.
underlying_name	String	Name of the resource packages in a queue.
owner	String	Owner of a resource package.

Example Request

None

Example Response

```
{
    "create_time": 1522055409139,
    "update_time": 1522228350501,
    "resource_type": "jar",
    "resource_name": "luxor-ommanager-dist.tar.gz",
    "status": "uploading",
    "underlying_name": "7885d26e-c532-40f3-a755-c82c442f19b8_luxor-ommanager-dist.tar.gz"
}
```


Status Codes

Table 17-39 describes the status code.

Table 17-39 Status codes

Status Codes	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.2.7 Deleting a Resource Package from a Group (Discarded)

Function

This API is used to delete resource packages in a group in a **Project**.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  DELETE /v2.0/{project_id}/resources/{resource_name}
- Parameter description

Table 17-40 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
resource_name	Yes	String	Name of the resource package that is uploaded.

Table 17-41 query parameter description

Parameter	Mandatory	Type	Description
group	No	String	Name of the package group returned when the resource package is uploaded.

NOTE

The following is an example of the URL containing the **query** parameter:
DELETE /v2.0/{project_id}/resources/{resource_name}*?group={group}*

Request

None

Response

- Code 200 is returned if you successfully delete a resource package.
- Code 404 is returned if you initiate a request to delete a resource package that does not exist.

Example Request

None

Example Response

None

Status Codes

Table 17-42 describes the status code.

Table 17-42 Status codes

Status Code	Description
200	Deletion succeeded.
404	Not found.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.2.8 Changing the Owner of a Group or Resource Package (Discarded)

Function

This API is used to change the owner of a program package.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  PUT /v2.0/{project_id}/resources/owner
- Parameter description

Table 17-43 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-44 Request parameters

Parameter	Mandatory	Type	Description
new_owner	Yes	String	New username. The name contains 5 to 32 characters, including only digits, letters, underscores (_), and hyphens (-). It cannot start with a digit.
group_name	Yes	String	Group name. The name contains a maximum of 64 characters. Only digits, letters, periods (.), underscores (_), and hyphens (-) are allowed.

Parameter	Mandatory	Type	Description
resource_name	No	String	Package name. The name can contain only digits, letters, underscores (_), exclamation marks (!), hyphens (-), and periods (.), but cannot start with a period. The length (including the file name extension) cannot exceed 128 characters. **This parameter is mandatory if you want to change the owner of a resource package in a group.**

NOTE

group_name and **resource_name** can be used independently or together.

- To change the owner of a group, use **group_name**.
- To change the owner of a resource package, use **resource_name**.
- To change the owner of a resource package in a group, use **group_name** and **resource_name** at the same time.

Response

Table 17-45 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Example Request

Change the group name of the program package to **groupName** and the user name to **scuser1**.

```
{
    "new_owner": "scuser1",
    "group_name": "groupName"
}
```

Example Response

```
{
    "is_success": "true",
    "message": ""
}
```


Status Codes

Table 17-46 describes the status code.

Table 17-46 Status codes

Status Code	Description
200	The modification operations are successful.
404	Request error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 17-47 Error codes

Error Code	Error Message
DLI.0002	No such user. userName:ssssss.

17.3 APIs Related to Spark Batch Processing (Discarded)

17.3.1 Querying Batch Job Logs (Discarded)

Function

This API is used to query the back-end logs of batch processing jobs.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  GET /v2.0/{project_id}/batches/{batch_id}/log
- Parameter description

Table 17-48 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
batch_id	Yes	String	ID of a batch processing job.

Table 17-49 Request parameters

Parameter	Mandatory	Type	Description
from	No	Integer	Start line of the log to be displayed. By default, the last 100 lines of the log are displayed. If a log file contains fewer than 100 lines, line 0 is the start line.
size	No	Integer	Number of logs to be queried.
type	No	String	If **type** is set to **driver**, the Spark Driver log is generated.
index	No	Integer	When a submitted job is retried, multiple driver logs are generated. This parameter specifies the **index** number of the specified driver log. The default value is **0**. This parameter must be used together with the **type** parameter. If only **index** is specified, the default value of **type** is **driver**.

Request

None

Response

Table 17-50 Response parameters

Parameter	Mandatory	Type	Description
id	No	String	ID of a batch processing job.
from	No	Integer	Start index of a log.
total	No	Long	Total number of records in a log.

Parameter	Mandatory	Type	Description
log	No	Array of Strings	Log of the current batch processing job.

Example Request

None

Example Response

```
{
    "id": "0a324461-d9d9-45da-a52a-3b3c7a3d809e",
    "from": 0,
    "total": 3,
    "log": [
        "Detailed information about job logs"
    ]
}
```

Status Codes

Table 17-51 describes the status code.

Table 17-51 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.4 SQL Job-related APIs (Discarded)

17.4.1 Importing Data (Discarded)

Function

This API is used to import data from a file to a DLI or OBS table. Currently, only OBS data can be imported to a DLI or OBS table.

NOTE

- This API has been discarded and is not recommended.
- This API is asynchronous.
- When importing data, you can select an existing OBS bucket path or create an OBS bucket path, but only one OBS bucket path can be specified.
- If you need to create an OBS bucket, ensure that the bucket name complies with the following naming rules:
  - The name must be globally unique in OBS.
  - The name must contain 3 to 63 characters. Only lowercase letters, digits, hyphens (-), and periods (.) are allowed.
  - The name cannot start or end with a period (.) or hyphen (-), and cannot contain two consecutive periods (.) or contain a period (.) and a hyphen (-) adjacent to each other.
  - The name cannot be an IP address.
  - If the name contains any period (.), the security certificate verification may be triggered when you access the bucket or objects in the bucket.
- If the type of a column in the source file to be imported does not match that of the target table, the query result of the row will be null.
- Two or more concurrent tasks of importing data to the same table are not allowed.

URI

- URI format

  POST /v1.0/{project_id}/jobs/import-table

- Parameter description

Table 17-52 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-53 Request parameters

Parameter	Mandatory	Type	Description
data_path	Yes	String	Path to the data to be imported. Currently, only OBS data can be imported.

Parameter	Mandatory	Type	Description
data_type	Yes	String	Type of the data to be imported. Currently, data types of CSV, Parquet, ORC, JSON, and Avro are supported. **NOTE** Data in **Avro** format generated by Hive tables cannot be imported.
database_name	Yes	String	Name of the database where the table to which data is imported resides.
table_name	Yes	String	Name of the table to which data is imported.
with_column_header	No	Boolean	Whether the first line of the imported data contains column names, that is, headers. The default value is **false**, indicating that column names are not contained. This parameter can be specified when CSV data is imported.
delimiter	No	String	User-defined data delimiter. The default value is a comma (,). This parameter can be specified when CSV data is imported.
quote_char	No	String	User-defined quotation character. The default value is double quotation marks ("). This parameter can be specified when CSV data is imported.
escape_char	No	String	User-defined escape character. The default value is a backslash (\). This parameter can be specified when CSV data is imported.
date_format	No	String	Specified date format. The default value is **yyyy-MM-dd**. For details about the characters involved in the date format, see **Table 17-54**. This parameter can be specified when data in the CSV or JSON format is imported.
bad_records_path	No	String	**Bad records** storage directory during job execution. After configuring this item, the **bad records** is not imported into the target table.
timestamp_format	No	String	Specified time format. The default value is **yyyy-MM-dd HH:mm:ss**. For definitions about characters in the time format, see **Table 17-54**. This parameter can be specified when data in the CSV or JSON format is imported.
queue_name	No	String	Name of the queue that is specified to execute a task. If no queue is specified, the default queue is used.

Parameter	Mandatory	Type	Description
overwrite	No	Boolean	Whether to overwrite data. The default value is **false**, indicating appending write. If the value is **true**, it indicates overwriting.
partition_spec	No	Object	Partition to which data is to be imported. • If this parameter is not set, the entire table data is dynamically imported. The imported data must contain the data in the partition column. • If this parameter is set and all partition information is configured during data import, data is imported to the specified partition. The imported data cannot contain data in the partition column. • If not all partition information is configured during data import, the imported data must contain all non-specified partition data. Otherwise, abnormal values such as **null** exist in the partition field column of non-specified data after data import.
conf	No	Array of Strings	User-defined parameter that applies to the job. Currently, **dli.sql.dynamicPartitionOverwrite.enabled** can be set to **false** by default. If it is set to **true**, data in a specified partition is overwritten. If it is set to **false**, data in the entire DataSource table is dynamically overwritten. **NOTE** For dynamic overwrite of Hive partition tables, only the involved partition data can be overwritten. The entire table data cannot be overwritten.

Table 17-54 Definition of characters involved in the date and time patterns

Character	Date or Time Element	Example
G	Epoch ID	AD
y	Year	1996; 96
M	Month	July; Jul; 07
w	Which week in a year	27 (Week 27 in the year)
W	Which week in a month	2 (Second week in the month)

Character	Date or Time Element	Example
D	Which day in a year	189 (Day 189 in the year)
d	Which day in a month	10 (Day 10 in the month)
u	Which day in a week	1 (Monday), ..., 7 (Sunday)
a	am/pm flag	pm (Afternoon)
H	Hour time (0-23)	2
h	Hour time (1-12)	12
m	Minute time	30
s	Second time	55
S	Which milliseconds	978
z	Time zone	Pacific Standard Time; PST; GMT-08:00

Response

Table 17-55 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully sent. Value **true** indicates that the request is successfully sent.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_id	No	String	ID of a job returned after a job is generated and submitted by using SQL statements. The job ID can be used to query the job status and results.
job_mode	No	String	Job execution mode. The options are as follows: ● **async**: asynchronous ● **sync**: synchronous

Example Request

Import the CSV data stored on OBS to **db2.t2**.

```
{
   "data_path": "obs://home/data1/DLI/t1.csv",
   "data_type": "csv",
   "database_name": "db2",
   "table_name": "t2",
   "with_column_header": false,
   "delimiter": ",",
   "quote_char": ",",
   "escape_char": ",",
   "date_format": "yyyy-MM-dd",
   "timestamp_format": "yyyy-MM-dd'T'HH:mm:ss.SSSZZ",
   "queue_name": "queue2",
   "overwrite": false,
   "partition_spec":{
    "column1":  "2020-01-01",
    "column2":  "columnPartValue"
    }
}
```

Example Response

```
{
  "is_success": true,
  "message": "import data to table t2 started",
  "job_id": "6b29eb77-4c16-4e74-838a-2cf7959e9202",
  "job_mode":"async"
}
```

Status Codes

Table 17-56 describes the status code.

Table 17-56 Status codes

Status Code	Description
200	Import succeeded.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.4.2 Exporting Data (Discarded)

Function

This API is used to export data from a DLI table to a file.

- This API is asynchronous.
- Currently, data can be exported only from a DLI table to OBS, and the OBS path must be specified to the folder level. The OBS path cannot contain commas (,). The OBS bucket name cannot end with the regular expression format **.[0-9]+(.*)**. Specifically, if the bucket name contains dots (.), the last dot (.) cannot be followed by a digit, for example, ****.12abc** and ****.12**.
- 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.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  POST /v1.0/{project_id}/jobs/export-table
- Parameter description

Table 17-57 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-58 Request parameters

Parameter	Mandatory	Type	Description
data_path	Yes	String	Path for storing the exported data. Currently, data can be stored only on OBS. If **export_mode** is set to **errorifexists**, the OBS path cannot contain the specified folder, for example, the **test** folder in the example request.
data_type	Yes	String	Type of data to be exported. Currently, only CSV and JSON are supported.
database_name	Yes	String	Name of the database where the table from which data is exported resides.
table_name	Yes	String	Name of the table from which data is exported.

Parameter	Mandatory	Type	Description
compress	Yes	String	Compression mode for exported data. Currently, the compression modes **gzip**, **bzip2**, and **deflate** are supported. If you do not want to compress data, enter **none**.
queue_name	No	String	Name of the queue that is specified to execute a task. If no queue is specified, the default queue is used.
export_mode	No	String	Export mode. The parameter value can be **ErrorIfExists** or **Overwrite**. If **export_mode** is not specified, this parameter is set to **ErrorIfExists** by default. • **ErrorIfExists**: Ensure that the specified export directory does not exist. If the specified export directory exists, an error is reported and the export operation cannot be performed. • **Overwrite**: If you add new files to a specific directory, existing files will be deleted.
with_column_header	No	Boolean	Whether to export column names when exporting CSV and JSON data. • If this parameter is set to **true**, the column names are exported. • If this parameter is set to **false**, the column names are not exported. • If this parameter is left blank, the default value **false** is used.

Response

Table 17-59 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully sent. Value **true** indicates that the request is successfully sent.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_id	No	String	ID of a job returned after a job is generated and submitted by using SQL statements. The job ID can be used to query the job status and results.

Parameter	Mandatory	Type	Description
job_mode	No	String	Job execution mode. The options are as follows: • **async**: asynchronous • **sync**: synchronous

Example Request

Export data from **db2.t2** to OBS and store the data in JSON format.

```
{
   "data_path": "obs://home/data1/DLI/test",
   "data_type": "json",
   "database_name": "db2",
   "table_name": "t2",
   "compress": "gzip",
   "with_column_header": "true",
   "queue_name": "queue2"
}
```

Example Response

```
{
  "is_success": true,
  "message": "export all data from table db2.t2 to path obs://home/data1/DLI/test started",
  "job_id": "828d4044-3d39-449b-b32c-957f7cfadfc9",
  "job_mode":"async"
}
```

Status Codes

Table 17-60 describes the status code.

Table 17-60 Status codes

Status Code	Description
200	Export successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.5 Resource-related APIs (Discarded)

17.5.1 Database-related APIs (Discarded)

17.5.1.1 Creating a Database (Discarded)

Function

This API is used to add a database.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  POST /v1.0/{project_id}/databases
- Parameter description

Table 17-61 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-62 Request parameters

Parameter	Mandatory	Type	Description
database_name	Yes	String	Name of the created database. • 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. **NOTE** The **default** database is a built-in database. You cannot create a database named **default**.
description	No	String	Information about the created database.

Parameter	Mandatory	Type	Description
enterprise_project_id	No	String	Enterprise project ID. The value **0** indicates the default enterprise project. **NOTE** Users who have enabled Enterprise Management can set this parameter to bind a specified project.
tags	No	Array of objects	Database tag. For details, see **Table 17-63**.

Table 17-63 tags parameters

Parameter	Mandatory	Type	Description
Key	Yes	String	Tag key. **NOTE** A tag key can contain a maximum of 36 characters. Special characters (=*<>|) are not allowed, and the key cannot start with a space.
value	Yes	String	Tag key. **NOTE** A tag value can contain a maximum of 43 characters. Special characters (=*<>|) are not allowed, and the value cannot start with a space.

Response

Table 17-64 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Example Request

Create a test database named **db1**.

```
{
  "database_name": "db1",
```



```
 "description": "this is for test"
}
```

Example Response

```
{
 "is_success": true,
 "message": ""
}
```

Status Codes

Table 17-65 describes the status code.

Table 17-65 Status codes

Status Code	Description
200	The job is created successfully.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.5.1.2 Deleting a Database (Discarded)

Function

This API is used to delete an empty database. If there are tables in the database to be deleted, delete all tables first. For details about the API used to delete tables, see **Deleting a Table (Discarded)**.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  DELETE /v1.0/{project_id}/databases/{database_name}
- Parameter description

Table 17-66 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
database_name	Yes	String	Name of the database to be deleted.

Table 17-67 query parameter description

Parameter	Mandatory	Type	Description
cascade	No	Boolean	Specifies whether to forcibly delete the database. The value can be **true** or **false**. Default value: **false**.
async	No	Boolean	Specifies whether to delete the database in asynchronous mode. The value can be **true** or **false**. Default value: **false**.

NOTE

The following is an example of the URL containing the **query** parameter:

DELETE /v1.0/{project_id}/databases/{database_name}?cascade=*{is_cascade}*&async=*{is_asyn}*

Request

None

Response

Table 17-68 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Parameter	Mandatory	Type	Description
job_id	No	String	Returned job ID, which can be used to obtain the job status and result.
job_type	No	String	Type of a job. The options are as follows: • DDL • DCL • IMPORT • EXPORT • QUERY • INSERT
job_mode	No	String	Job execution mode. The options are as follows: • **async**: asynchronous • **sync**: synchronous

Example Request

None

Example Response

- The following is an example of a successful response in synchronous mode:

```
{
 "is_success": true,
 "message": "",
 "job_mode": "sync"
}
```

- The following is an example of a successful response in asynchronous mode:

```
{
 "is_success": true,
 "message": "",
 "job_id": "208b08d4-0dc2-4dd7-8879-ddd4c020d7aa",
 "job_type": "DDL",
 "job_mode": "async"
}
```

NOTE

- If the database is deleted asynchronously, you can view the current job status by calling the API for querying job status. For details, see **Querying Job Status**.
- If **cascade** is set to **true**, all tables in the database will be deleted. Exercise caution when performing this operation.

Status Codes

Table 17-69 describes the status code.

Table 17-69 Status codes

Status Code	Description
200	Deletion succeeded.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.5.1.3 Querying All Databases (Discarded)

Function

This API is used to query the information about all the databases.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  GET /v1.0/{project_id}/databases
- Parameter description

Table 17-70 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 17-71 query parameter description

Parameter	Mandatory	Type	Description
with-priv	No	Boolean	Specifies whether to display the permission information. The value can be **true** or **false**. The default value is **false**.

Parameter	Mandatory	Type	Description
offset	No	Integer	The value should be no less than **0**. The default value is **0**.
limit	No	Integer	Number of returned data records. The value must be greater than or equal to **0**. By default, all data records are returned.
keyword	No	String	Database name filtering keyword. Fuzzy match is used to obtain all databases whose names contain the keyword.
tags	No	String	Database tags. The format is **key=value**. ● Request with one specified tag GET /v1.0/{project_id}/databases?offset=0&limit=10&with-priv=true&tags=k1%3Dv1 The equal sign (=) is escaped to **%3D**, **k1** indicates the tag key, and **v1** indicates the tag value. ● Request with more than one tag Use commas (,) to separate tags. The commas (,) must be escaped to **%2C**. For example: GET /v1.0/{project_id}/databases?offset=0&limit=10&with-priv=true&tags=k1%3Dv1%2Ck2%3Dv2 The equal sign (=) is escaped to **%3D**. **k1** indicates a tag key, and **v1** indicates the tag value. **k2** indicates another tag key, and **v2** indicates the tag value. Currently, only fuzzy query is supported. Exact query is not supported.

NOTE

The following is an example of the URL containing the **query** parameter:

GET /v1.0/{project_id}/databases?with-priv=*{is_with_priv}*&offset=*{offsetValue}*&limit=*{limitValue}*&keyword=*{keywordValue}*
?tags=*{tagsValue}*

Request

None

Response

Table 17-72 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
database_count	No	Integer	Total number of databases.
databases	No	Array of objects	Database information. For details, see **Table 17-73**.

Table 17-73 databases parameters

Parameter	Mandatory	Type	Description
database_name	No	String	Name of a database.
owner	No	String	Creator of a database.
table_number	No	Integer	Number of tables in a database.
description	No	String	Information about a database.
enterprise_project_id	Yes	String	Enterprise project ID. The value **0** indicates the default enterprise project. **NOTE** Users who have enabled Enterprise Management can set this parameter to bind a specified project.
resource_id	Yes	String	Resource ID.

Example Request

None

Example Response

```
{
  "is_success": true,
  "message": "",
  "database_count": 1,
  "databases": [
```



```
  {
    "database_name": "db2",
    "description": "this is for test",
    "owner": "tenant1",
    "table_number": 15

  }
 ]
}
```

Status Codes

Table 17-74 describes the status code.

Table 17-74 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.5.1.4 Modifying a Database Owner (Discarded)

Function

This API is used to modify the owner of a database.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  PUT /v1.0/{project_id}/databases/{database_name}/owner
- Parameter description

Table 17-75 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
database_name	Yes	String	Name of a database.

Request

Table 17-76 Request parameters

Parameter	Mandatory	Type	Description
new_owner	Yes	String	Name of the new owner. The new user must be a sub-user of the current tenant.

Response

Table 17-77 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Example Request

Change the owner of the database to **scuser1**.

```
{
    "new_owner": "scuser1"
}
```

Example Response

```
{
  "is_success": true,
```



```
 "message": ""
}
```

Status Codes

Table 17-78 describes the status code.

Table 17-78 Status codes

Status Code	Description
200	The modification operations are successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.5.2 Table-related APIs (Discarded)

17.5.2.1 Creating a Table (Discarded)

Function

This API is used to create a table. This API is a synchronous API.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  POST /v1.0/{project_id}/databases/{database_name}/tables
- Parameter description

Table 17-79 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Parameter	Mandatory	Type	Description
database_name	Yes	String	Name of the database where the new table resides.

Request

Table 17-80 Request parameters

Parameter	Mandatory	Type	Description
table_name	Yes	String	Name of the created table. • 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 ($). Example: **$test** • The length of the database name cannot exceed 128 characters.
data_location	Yes	String	Location where data is stored. The options are as follows: • OBS: OBS table • DLI: DLI table • VIEW: VIEW table
description	No	String	Information about the new table.
columns	Yes	Array of Objects	Columns of the new table. For details about column parameters, see **Table 17-82**. This parameter is optional when **data_location** is **VIEW**.
select_statement	No	String	Query statement required for creating a view. The database to which the table belongs needs to be specified in the query statement, in the format of *database.table*. This parameter is mandatory when **data_location** is **VIEW**.
data_type	No	String	Type of the data to be added to the OBS table. The options are as follows: Parquet, ORC, CSV, JSON, and Avro. **NOTE** This parameter is mandatory for an OBS table.

Parameter	Mandatory	Type	Description
data_path	No	String	Storage path of data in the new OBS table, which must be a path on OBS and must begin with **obs**. **NOTE** This parameter is mandatory for an OBS table. Do not set this parameter to the OBS root directory. Otherwise, all data in the root directory will be cleared when you clear table data.
with_column_header	No	Boolean	Whether the table header is included in the OBS table data. Only data in CSV files has this attribute. This parameter is mandatory when **data_location** is **OBS**.
delimiter	No	String	User-defined data delimiter. Only data in CSV files has this attribute. This parameter is mandatory when **data_location** is **OBS**.
quote_char	No	String	User-defined reference character. Double quotation marks ("\") are used by default. Only data in CSV files has this attribute. This parameter is mandatory when **data_location** is **OBS**.
escape_char	No	String	User-defined escape character. Backslashes (\\) are used by default. Only data in CSV files has this attribute. This parameter is mandatory when **data_location** is **OBS**.
date_format	No	String	User-defined date type. **yyyy-MM-dd** is used by default. For details about the characters involved in the date format, see **Table 17-54**. Only data in CSV and JSON files has this attribute. This parameter is mandatory when **data_location** is **OBS**.
timestamp_format	No	String	User-defined timestamp type. **yyyy-MM-dd HH:mm:ss** is used by default. For definitions about characters in the timestamp format, see **Table 17-54**. Only data in CSV and JSON files has this attribute. This parameter is mandatory when **data_location** is **OBS**.
tags	No	Array of objects	Database tag. For details about this object, see **tags parameters**.

Table 17-81 tags parameters

Parameter	Mandatory	Type	Description
key	Yes	String	Tag key **NOTE** A tag key can contain a maximum of 36 characters. Special characters (=*<>|) are not allowed, and the key cannot start with a space.
value	Yes	String	Tag value **NOTE** A tag value can contain a maximum of 43 characters. Special characters (=*<>|) are not allowed, and the value cannot start with a space.

Table 17-82 columns parameters

Parameter	Mandatory	Type	Description
column_name	Yes	String	Name of a column.
type	Yes	String	Data type of a column.
description	No	String	Description of a column.
is_partition_column	No	Boolean	Whether the column is a partition column. The value **true** indicates a partition column, and the value **false** indicates a non-partition column. The default value is **false**. **NOTE** When creating a partition table, ensure that at least one column in the table is a non-partition column. For details, see "Request example".

Response

Table 17-83 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.

Parameter	Mandatory	Type	Description
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Example Request

- Create a table whose **data_location** is **OBS** and data format of CSV.

```
{
 "table_name": "tb1",
 "data_location": "OBS",
 "description": "",
 "data_type": "csv",
 "data_path": "obs://obs/path1",
 "columns": [
 {
    "column_name": "column1",
    "type": "string",
    "description": "",
    "is_partition_column": true
 },
 {
    "column_name": "column2",
    "type": "string",
    "description": "",
    "is_partition_column": false
 }
 ],
 "with_column_header": true,
 "delimiter": ",",
 "quote_char": "\"",
 "escape_char": "\\",
 "date_format": "yyyy-MM-dd",
 "timestamp_format": "yyyy-MM-dd HH:mm:ss"
}
```

  NOTE

  The values of **date_format** and **timestamp_format** must be the same as the time format in the imported CSV file.

- Create a table whose **data_location** is **DLI**.

```
{
 "table_name": "tb2",
 "data_location": "DLI",
 "columns": [
  {
    "column_name": "column1",
    "type": "string",
    "description": "",
    "is_partition_column": true
  },
  {
    "column_name": "column2",
    "type": "string",
    "description": "",
    "is_partition_column": false
  }
 ],
 "tags": [
  {
```



```
    "key": "quarterly",
    "value": "Q3"
  },
  {
    "key": "author",
    "value": "user"
  }
 ]
}
```

- Create a table whose **data_location** is **VIEW**.

```
{
 "table_name": "tb3",
 "data_location": "VIEW",
 "columns": [
 {
    "column_name": "column1",
    "type": "string",
    "description": "",
    "is_partition_column": true
 },
 {
    "column_name": "column2",
    "type": "string",
    "description": "",
    "is_partition_column": false
 }
 ],
 "select_statement": "select * from db1.tb1"
}
```

Example Response

```
{
  "is_success": true,
  "message": ""
}
```

Status Codes

Table 17-84 describes the status code.

Table 17-84 Status codes

Status Code	Description
200	The job is created successfully.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.5.2.2 Deleting a Table (Discarded)

Function

This API is used to delete a specified table.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  DELETE /v1.0/{project_id}/databases/{database_name}/tables/{table_name}
- Parameter description

Table 17-85 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
database_name	Yes	String	Name of the database where the table to be deleted resides.
table_name	Yes	String	Name of the table to be deleted.

Table 17-86 query parameter description

Parameter	Mandatory	Type	Description
async	No	Boolean	Specifies whether to delete the database in asynchronous mode. The value can be **true** or **false**. Default value: **false**.

NOTE

The following is an example of the URL containing the **query** parameter:

DELETE /v1.0/{project_id}/databases/{database_name}/tables/{table_name}?async=*{is_async}*

Request

None

Response

Table 17-87 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
job_mode	No	String	Job execution mode. The options are as follows: • **async**: asynchronous • **sync**: synchronous

Example Request

None

Example Response

```
{
 "is_success": true,
 "message": ""
}
```

Status Codes

Table 17-88 describes the status code.

Table 17-88 Status codes

Status Code	Description
200	Deletion succeeded.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.5.2.3 Querying All Tables (Discarded)

Function

This API is used to query information about tables that meet the filtering criteria or all the tables in the specified database.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  GET /v1.0/{project_id}/databases/{database_name}/tables
- Parameter description

Table 17-89 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
database_name	Yes	String	Name of the database where the table resides.

Table 17-90 query parameter description

Parameter	Mandatory	Type	Description
keyword	No	String	Keywords used to filter table names.
with-detail	No	Boolean	Whether to obtain detailed information about tables (such as owner and size). The default value is **false**.
page-size	No	Integer	Paging size. The minimum value is **1** and the maximum value is **100**.
current-page	No	Integer	Current page number. The minimum value is **1**.
with-priv	No	Boolean	Whether to return permission information.

Parameter	Mandatory	Type	Description
table-type	No	String	Database table type. The options are as follows: • **MANAGED_TABLE**: DLI table • **EXTERNAL_TABLE**: OBS table • **VIRTUAL_VIEW**: view
datasource-type	No	String	Data source type. The options are as follows: • CloudTable • CSS • DLI • GaussDB(DWS) • Geomesa • HBase • JDBC • Mongo • OBS • ODPS • OpenTSDB • Redis • RDS
without-tablemeta	No	Boolean	Whether to obtain the metadata of a table. The default value is **false**. If this parameter is set to **true**, the response speed can be greatly improved.

NOTE

The following is an example of the URL containing the **query** parameter:

GET /v1.0/{project_id}/databases/{database_name}/tables*?keyword=tb&with-detail=true*

Request

None

Response

Table 17-91 Response parameters

Parameter	Mandatory	Type	Description
is_success	Yes	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	Yes	String	System prompt. If execution succeeds, the parameter setting may be left blank.
table_count	Yes	Integer	Total number of tables.
tables	Yes	Array of objects	Table information. For details, see **Table 17-92**.

Table 17-92 tables parameters

Parameter	Mandatory	Type	Description
create_time	Yes	Long	Time when a table is created. The timestamp is expressed in milliseconds.
data_type	No	String	Type of the data to be added to the OBS table. The options are as follows: Parquet, ORC, CSV, JSON, and Avro. **NOTE** This parameter is available only for OBS tables.
data_location	Yes	String	Data storage location, which can be DLI or OBS.
last_access_time	Yes	Long	Time when the table was last updated. The timestamp is expressed in milliseconds.
location	No	String	Storage path on the OBS table. **NOTE** This parameter is available only for OBS tables.
owner	Yes	String	Table owner.
table_name	Yes	String	Name of a table.
table_size	Yes	Long	Size of a DLI table. Set parameter to **0** for non-DLI tables. The unit is byte.

Parameter	Mandatory	Type	Description
table_type	Yes	String	Type of a table. • **EXTERNAL**: Indicates an OBS table. • **MANAGED**: Indicates a DLI table. • **VIEW**: Indicates a view.
partition_columns	No	Array of Strings	Partition field. This parameter is valid only for OBS partition tables.
page-size	No	Integer	Paging size. The minimum value is **1** and the maximum value is **100**.
current-page	No	Integer	Current page number. The minimum value is **1**.

NOTE

If **with-detail** is set to **false** in the URI, only values of tables-related parameters **data_location**, **table_name**, and **table_type** are returned.

Example Request

None

Example Response

```
{
  "is_success": true,
  "message": "",
  "table_count": 1,
  "tables": [
    { "create_time":1517364268000,
      "data_location":"OBS",
      "data_type":"csv",
      "last_access_time":1517364268000,
      "location":"obs://DLI/sqldata/data.txt",
      "owner":"test",
      "partition_columns": ["a0"],
      "table_name":"obs_t",
      "table_size":0,
      "table_type":"EXTERNAL"
    }
  ]
}
```

Status Codes

Table 17-93 describes the status code.

Table 17-93 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.5.2.4 Describing Table Information (Discarded)

Function

This API is used to describe metadata information in the specified table.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  GET /v1.0/{project_id}/databases/{database_name}/tables/{table_name}

- Parameter description

Table 17-94 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
database_name	Yes	String	Name of the database where the target table resides.
table_name	Yes	String	Name of the target table.

Request

None

Response

Table 17-95 Response parameters

Parameter	Mandatory	Type	Description
is_success	Yes	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	Yes	String	System prompt. If execution succeeds, the parameter setting may be left blank.
column_count	Yes	Integer	Total number of columns in the table.
columns	Yes	Array of objects	Column information, including the column name, type, and description. For details, see **Table 17-96**.
table_type	Yes	String	Table type. The options are as follows: **MANAGED**: DLI table **EXTERNAL**: OBS table **VIEW**: view
data_type	No	String	Data type, including **CSV**, **Parquet**, **ORC**, **JSON**, and **Avro**.
data_location	No	String	Path for storing data, which is an OBS path.
storage_properties	No	Array of objects	Storage attribute, which is in the format of **key/value** and includes parameters **delimiter**, **escape**, **quote**, **header**, **dateformat**, and **timestampformat**.
table_comment	No	String	Table comment.
create_table_sql	No	String	Statement used to create a table.

Table 17-96 columns parameters

Parameter	Mandatory	Type	Description
column_name	Yes	String	Column name.
description	Yes	String	Description of a column.

Parameter	Mandatory	Type	Description
type	Yes	String	Data type of a column.
is_partition_column	Yes	Boolean	Indicates whether the column is a partition column. The value **true** indicates that the column is a partition column, and the value **false** indicates that the column is not a partition column. The default value is **false**.

Example Request

None

Example Response

- **MANAGED** type table

```
{
  "is_success": true,
  "message": "",
  "column_count": 3,
  "columns": [
    {
      "column_name": "id",
      "description": "",
      "type": "int",
      "is_partition_column": false
    },
    {
      "column_name": "name",
      "description": "",
      "type": "string",
      "is_partition_column": false
    },
    {
      "column_name": "level",
      "description": "",
      "type": "string",
      "is_partition_column": true
    }
  ],
    "table_type":"MANAGED"
}
```

- **EXTERNAL** type table

```
{
    "is_success": true,
    "message": "",
    "column_count": 2,
    "columns": [
        {
            "type": "string",
            "description": "",
            "column_name": "col2",
            "is_partition_column": false
        },
        {
            "type": "string",
            "description": "",
```



```
            "column_name": "col1",
            "is_partition_column": true
        }
    ],
    "table_type": "EXTERNAL",
    "data_type": "parquet",
    "data_location": "obs://obs-wangtao/savepoint/savepoint-d95437-039668840fff/_metadata",
    "storage_properties": [
        {
            "key": "timestampformat",
            "value": "yyyy-MM-dd HH:mm:ss"
        },
        {
            "key": "quote",
            "value": "\""
        },
        {
            "key": "dateformat",
            "value": "yyyy-MM-dd"
        },
        {
            "key": "escape",
            "value": "\\"
        },
        {
            "key": "header",
            "value": "false"
        },
        {
            "key": "delimiter",
            "value": ","
        }
    ],
    "table_comment": "",
     "create_table_sql": "CREATE TABLE `default`.`wan_test` (`col2` STRING, `col1` STRING)\nUSING
parquet\nOPTIONS (\n  `timestampformat` 'yyyy-MM-dd HH:mm:ss',\n  `quote` '\"',\n  `dateformat`
'yyyy-MM-dd',\n  `escape` '\\\\',\n  `header` 'false',\n  `delimiter` ',')\nPARTITIONED BY
(col1)\nCOMMENT ''\nLOCATION 'obs://obs-wangtao/savepoint/savepoint-d95437-039668840fff/
_metadata'\nTBLPROPERTIES (\n  'hive.serialization.extend.nesting.levels' = 'true'\n)\n"
  }
```


- **VIEW** type table


```
{
  "is_success": true,
  "message": "",
  "column_count": 3,
  "columns": [
    {
      "column_name": "id",
      "description": "",
      "type": "int",
      "is_partition_column": false
    },
    {
      "column_name": "name",
      "description": "",
      "type": "string",
      "is_partition_column": false
    },
    {
      "column_name": "level",
      "description": "",
      "type": "string",
      "is_partition_column": true
    }
  ],
  "table_type":"VIEW",
  "create_table_sql": "CREATE VIEW `default`.`view1`(id, name) AS\nselect * from a_gff.testtable\n"
}
```


Status Codes

Table 17-97 describes the status code.

Table 17-97 Status codes

Status Code	Description
200	The operation is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.5.2.5 Previewing Table Content (Discarded)

Function

This API is used to preview the first 10 rows in a table.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  GET /v1.0/{project_id}/databases/{database_name}/tables/{table_name}/preview

- Parameter description

Table 17-98 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
database_name	Yes	String	Name of the database where the table to be previewed resides.
table_name	Yes	String	Name of the table to be previewed.

Table 17-99 query parameter description

Parameter	Mandatory	Type	Description
mode	No	String	Preview table mode. The options are **SYNC** and **ASYNC**. The default value is **SYNC**.

NOTE

The following is an example of the URL containing the **query** parameter:

GET /v1.0/{project_id}/databases/{database_name}/tables/{table_name}/preview?mode=*{previewMode}*

Request

None

Response

Table 17-100 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
schema	No	Array of Map	Column name and type of a table.
rows	No	Array of objects	Previewed table content.

Example Request

None

Example Response

The following is an example of a successful response in synchronous mode:

```
{
    "is_success": true,
    "message": "",
    "schema": [
        {
          "id": "int"
```



```
      },
      {
        "name": "string"
      },
      {
        "address": "string"
      }
    ],
    "rows": [
      [
        "1",
        "John",
        "xxx"
      ],
      [
        "2",
        "Lily",
        "xxx"
      ]
    ]
}
```


NOTE

In asynchronous request mode, a job ID is returned. You can obtain the preview information based on the job ID.

Status Codes

Table 17-101 describes the status code.

Table 17-101 Status codes

Status Code	Description
200	The query is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.5.2.6 Listing Partitions (Discarded)

Function

This API is used to list partitions.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  GET /v1.0/{project_id}/databases/{database_name}/tables/{table_name}/partitions

- Parameter description

Table 17-102 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
database_name	Yes	String	Name of a database.
table_name	Yes	String	Name of a table.

Table 17-103 query parameter description

Parameter	Mandatory	Type	Description
limit	No	Integer	Number of returned records displayed on each page. The default value is **100**.
offset	No	Integer	Offset.
filter	No	String	Filtering condition. Currently, only the = condition is supported. For example, **name=name1** indicates that the data whose name is **name1** in the partition is filtered. **name** indicates the name of the partition column, and **name1** indicates the value of the partition column. The key and value are case insensitive. Example: GET /v1.0/{*project_id*}/databases/{*database_name*}/tables/{*table_name*}/partitions?part=part2

Request

None

Response

Table 17-104 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
partitions	No	Object	Partition information. For details, see **Table 17-105**.

Table 17-105 partitions parameter description

Parameter	Mandatory	Type	Description
total_count	Yes	Long	Total number of partitions.
partition_infos	Yes	Array of objects	List of partitions. For details, see **Table 17-106**.

Table 17-106 partition_infos parameter description

Parameter	Mandatory	Type	Description
partition_name	Yes	String	Partition name.
create_time	Yes	Long	Time when a partition is created.
last_access_time	Yes	Long	Last update time.
locations	No	Array of Strings	Path.
last_ddl_time	No	Long	Execution time of the last DDL statement, in seconds.
num_rows	No	Long	Total rows in the partition.
num_files	No	Long	Number of files in a partition.
total_size	No	Long	Total size of data in the partition, in bytes.

Example Request

None

Example Response


```
{
    "is_success": true,
    "message": "list partitions succeed",
    "partitions": {
        "total_count": 5,
        "partition_infos": [
            {
                "partition_name": "name=test",
                "create_time": 1579520179000,
                "last_access_time": 1579520179000,
                "locations": [
                    "obs://test/partition"
                ]
            },
            {
                "partition_name": "name=test1",
                "create_time": 1579521406000,
                "last_access_time": 1579521406000,
                "locations": [
                    "obs://test/partition"
                ]
            },
            {
                "partition_name": "name=test2",
                "create_time": 1579521884000,
                "last_access_time": 1579521884000,
                "locations": [
                    "obs://test/partition"
                ]
            },
            {
                "partition_name": "name=test3",
                "create_time": 1579522085000,
                "last_access_time": 1579522085000,
                "locations": [
                    "obs://test/partition"
                ]
            },
            {
                "partition_name": "name=name1/age=age1",
                "create_time": 1581409182000,
                "last_access_time": 1581409182000,
                "locations": [
                    "obs://test/0117"
                ],
                "last_ddl_time": 1581409182,
                "total_size": 2130,
                "num_rows": -1,
                "num_files": 2
            }
        ]
    }
}
```


Status Codes

Table 17-107 describes the status code.

Table 17-107 Status codes

Status Code	Description
200	The operation is successful.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.6 Permission-related APIs (Discarded)

17.6.1 Granting Queue Permissions to a User (Discarded)

Function

This API is used to share a specific queue with other users. You can grant users with the permission to use the specified queue or revoke the permission.

The user group the user belongs to must have the **Tenant Guest** permission in the region where the user group belongs.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  PUT /v1.0/{project_id}/queues/user-authorization
- Parameter description

Table 17-108 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-109 Request parameters

Parameter	Mandatory	Type	Description
queue_name	Yes	String	Name of a queue. Example value: **queue1**.
user_name	Yes	String	Name of the user who is granted with usage permission on a queue or whose queue usage permission is revoked or updated. Example value: **tenant2**.
action	Yes	String	Grants or revokes the permission. The parameter value can be **grant**, **revoke**, or **update**. Users can perform the **update** operation only when they have been granted with the **grant** and **revoke** permissions. Example value: **grant**. • **grant**: Indicates to grant users with permissions. • **revoke**: Indicates to revoke permissions. • **update**: Indicates to clear all the original permissions and assign the permissions in the provided permission array.
privileges	Yes	Array of Strings	List of permissions to be granted, revoked, or updated. The following permissions are supported: Example value: [**DROP_QUEUE**, **SUBMIT_JOB**]. • **SUBMIT_JOB**: indicates to submit a job. • **CANCEL_JOB**: indicates to cancel a job. • **DROP_QUEUE**: indicates to a delete a queue. • **GRANT_PRIVILEGE**: indicates to assign a permission. • **REVOKE_PRIVILEGE**: indicates to revoke a permission. • **SHOW_PRIVILEGES**: indicates to view the permissions of other users • **RESTART**: indicates to restart the queue. • **SCALE_QUEUE**: indicates to change the queue specifications. **NOTE** If the update list is empty, all permissions of the queue granted to the user are revoked.

Response

Table 17-110 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed. Example value: **true**.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank. Example value: left blank.

Example Request

Grant a user the permission to submit jobs on queue1 and delete queue1.

```
{
    "queue_name": "queue1",
    "user_name": "tenant2",
    "action": "grant",
    "privileges" : ["DROP_QUEUE", "SUBMIT_JOB"]
}
```

Example Response

```
{
  "is_success": true,
  "message": ""
}
```

Status Codes

Table 17-111 describes the status code.

Table 17-111 Status codes

Status Codes	Description
200	Authorization succeeds.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.6.2 Querying Queue Users (Discarded)

Function

This API is used to query names of all users who can use a specified queue.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  GET /v1.0/{project_id}/queues/{queue_name}/users
- Parameter description

Table 17-112 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
queue_name	Yes	String	Name of a queue.

Table 17-113 query parameter description

Parameter	Mandatory	Type	Description
limit	Yes	Integer	Number of records to be displayed of the page-based query.
offset	Yes	Integer	Specifies the offset of the page-based query.

Request

None

Response

Table 17-114 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed. Example value: **true**.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank. Example value: left blank.
queue_name	No	String	Name of a queue. Example value: **queue1**.
privileges	No	Array of Object	Users who are granted with the permission to use this queue and the permission array to which users belong. For details, see **Table 17-115**.
count	No	Long	Total number of users

Table 17-115 privileges parameters

Parameter	Mandatory	Type	Description
is_admin	No	Boolean	Whether the database user is an administrator. Example value: **false**.
user_name	No	String	Name of the user who has permission on the current queue. Example value: **user2**.
privileges	No	Array of Strings	Permission of the user on the queue. Example value: [**SUBMIT_JOB**].

Example Request

None

Example Response

```
{
  "is_success": true,
  "message": "",
  "count": 2,
  "privileges": [
    {
```



```
    "is_admin": true,
    "privileges": [
      "ALL"
    ],
    "user_name": "tenant1"
  },
  {
    "is_admin": false,
    "privileges": [
      "SUBMIT_JOB"
    ],
    "user_name": "user2"
  }
 ],
 "queue_name": "queue1"
}
```

Status Codes

Table 17-116 describes the status code.

Table 17-116 Status codes

Status Code	Description
200	Authorization succeeds.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.6.3 Granting Data Permission to Users (Discarded)

Function

This API is used to grant database or table data usage permission to specified users.

The user group containing the authorized user must have the **Tenant Guest** permission in the region where the user group belongs.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  PUT /v1.0/{project_id}/user-authorization

- Parameter description

Table 17-117 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-118 Request parameters

Parameter	Mandatory	Type	Description
user_name	Yes	String	Name of the user who is granted with usage permission on a queue or whose queue usage permission is revoked or updated. Example value: **user2**.
action	Yes	String	Grants or revokes the permission. The parameter value can be **grant**, **revoke**, or **update**. Example value: **grant**. • **grant**: Indicates to grant users with permissions. • **revoke**: Indicates to revoke permissions. • **update**: Indicates to clear all the original permissions and assign the permissions in the provided permission array. **NOTE** Users can perform the **update** operation only when they have been granted with the **grant** and **revoke** permissions.
privileges	Yes	Array of objects	Permission granting information. For details, see **Table 17-119**. Example value: [{"object": "databases.db1.tables.tb2.columns.column1","privileges": ["SELECT"]},"object": "databases.db1.tables.tbl","privileges": ["DROP_TABLE"]

Table 17-119 privileges parameters

Parameter	Mandatory	Type	Description
object	Yes	String	Data objects to be assigned. If they are named: • **databases.***Database name*, data in the entire database will be shared. • **databases.***Database name***.tables.***Table name*, data in the specified table will be shared. • **databases.***Database name***.tables.***Table name***.columns.***Column name*, data in the specified column will be shared. • **jobs.flink.***Flink job ID*, data the specified job will be shared. • **groups.***Package group name*, data in the specified package group will be shared. • **resources.***Package name*, data in the specified package will be shared. Example value: **databases.db1.tables.tb2.columns.column1**.
privileges	Yes	Array of Strings	List of permissions to be granted, revoked, or updated. Example value: [**SELECT**]. **NOTE** If **Action** is **Update** and the update list is empty, all permissions of the user in the database or table are revoked.

Response

Table 17-120 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed. Example value: **true**.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank. Example value: left blank.

Example Request

Grant **user2** the permission to query data in the database **db1**, delete the data table **db1.tbl**, and query data in a specified column **db1.tbl.column1** of a data table.


```
{
  "user_name": "user2",
  "action": "grant",
  "privileges": [
    {
      "object": "databases.db1.tables.tb2.columns.column1",
      "privileges": [
        "SELECT"
      ]
    },
    {
      "object": "databases.db1.tables.tbl",
      "privileges": [
        "DROP_TABLE"
      ]
    },
    {
      "object": "databases.db1",
      "privileges": [
        "SELECT"
      ]
    }
  ]
}
```

Example Response

```
{
  "is_success": true,
  "message": ""
}
```

Status Codes

Table 17-121 describes the status code.

Table 17-121 Status codes

Status Code	Description
200	Authorization succeeds.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.6.4 Querying Database Users (Discarded)

Function

This API is used query names of all users who have permission to use or access the database.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  GET /v1.0/{project_id}/databases/{database_name}/users
- Parameter description

Table 17-122 URI parameters

Parameter	Mandatory	Description
project_id	Yes	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
database_name	Yes	Name of the database to be queried.

Request

None

Response

Table 17-123 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed. Example value: **true**.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank. Example value: left blank.
database_name	No	String	Name of the database to be queried. Example value: **dsstest**.
privileges	No	Array of objects	Permission information. For details, see **Table 17-124**.

Table 17-124 privileges parameters

Parameter	Mandatory	Type	Description
is_admin	No	Boolean	Whether the database user is an administrator. Example value: **true**.
user_name	No	String	Name of the user who has permission on the current database. Example value: **test**.
privileges	No	Array of Strings	Permission of the user on the database. For details, see **Data Permissions List**. Example value: [**ALTER_TABLE_ADD_PARTITION**].

Example Request

None

Example Response

```
{
  "is_success": true,
  "message": "",
  "database_name": "dsstest",
  "privileges": [
    {
      "is_admin": true,
      "privileges": [
        "ALL"
      ],
      "user_name": "test"
    },
    {
      "is_admin": false,
      "privileges": [
        "ALTER_TABLE_ADD_PARTITION"
      ],
      "user_name": "scuser1"
    },
    {
      "is_admin": false,
      "privileges": [
        "CREATE_TABLE"
      ],
      "user_name": "scuser2"
    }
  ]
}
```

NOTE

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.6.5 Querying Table Users (Discarded)

Function

This API is used to query users who have permission to access the specified table or column in the table.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  GET /v1.0/{project_id}/databases/{database_name}/tables/{table_name}/users
- Parameter description

Table 17-125 URI parameters

Parameter	Mandatory	Description
project_id	Yes	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
database_name	Yes	Name of the database where the table to be queried resides.
table_name	Yes	Name of a table that is to be queried.

Request

None

Response

Table 17-126 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
privileges	No	Array <Objects>	Permission information. For details, see **Table 17-127**.

Table 17-127 privileges parameters

Parameter	Mandatory	Type	Description
is_admin	No	Boolean	Determines whether a user is an administrator. The value **false** indicates that the user is not an administrator, and the value **true** indicates that the user is an administrator.
object	No	String	Objects on which a user has permission. • If the object is in the format of **databases.*Database name*.tables.*Table name***, the user has permission on the database. • If the object is in the format of **databases.*Database name*.tables.*Table name*columns.*Column name***, the user has permission on the table.
privileges	No	Array<String>	Permission of the user on the object.
user_name	No	String	Name of the user who has the permission.

Example Request

None

Example Response

```
{
  "is_success": true,
  "message": "",
  "privileges": [
    {
      "is_admin": false,
      "object": "databases.dsstest.tables.csv_par_table",
      "privileges": [
        "SELECT"
      ],
      "user_name": "tent2"
    },
    {
      "is_admin": true,
      "object": "databases.dsstest.tables.csv_par_table",
      "privileges": [
        "ALL"
      ],
      "user_name": "tent4"
    }
  ]
}
```


NOTE

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.6.6 Querying a User's Table Permissions (Discarded)

Function

This API is used to query the permission of a specified user on a table.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  GET /v1.0/{project_id}/databases/{database_name}/tables/{table_name}/users/{user_name}

- Parameter description

  Table 17-128 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
database_name	Yes	String	Name of the database where the table to be queried resides.
table_name	Yes	String	Name of a table that is to be queried.
user_name	Yes	String	Name of the user whose permission is to be queried.

Request

None

Response

Table 17-129 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed. Example value: **true**.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank. Example value: left blank.
user_name	No	String	Name of the user whose permission is to be queried.
privileges	No	Array Of objects	Permission information. For details, see **Table 17-130**.

Table 17-130 privileges parameters

Parameter	Mandatory	Type	Description
object	No	String	Objects on which a user has permission. • If the object is in the format of **databases.*Database name*.tables.*Table name***, the user has permission on the database. • If the object is in the format of **databases.*Database name*.tables.*Table name*columns.*Column name***, the user has permission on the table. Example value: **databases.dsstest.tables.obs_231**.
privileges	No	Array of Strings	Permission of the user on a specified object. Example value: [**DESCRIBE_TABLE**]. For details about table permissions, see **Permissions Management**.

Example Request

None

Example Response

```
{
  "is_success": true,
```



```
 "message": "",
 "privileges": [
  {
    "object": "databases.dsstest.tables.obs_2312",
    "privileges": [
      "DESCRIBE_TABLE"
    ]
  },
  {
    "object": "databases.dsstest.tables.obs_2312.columns.id",
    "privileges": [
      "SELECT"
    ]
  }
 ],
 "user_name": "scuser1"
}
```


Status Codes

Table 17-131 describes the status code.

Table 17-131 Status codes

Status Code	Description
200	Authorization succeeds.
400	Request error.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.7 Queue-related APIs (Discarded)

17.7.1 Creating a Scheduled CU Change (Discarded)

Function

This API is used to create a scheduled CU change, that is, to create a scheduled CU change for a specified queue.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  POST /v1/{project_id}/queues/{queue_name}/plans

- Parameter description

Table 17-132 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
queue_name	Yes	String	Name of the queue for which you want to set a scheduled scaling plan. The name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_). The name contains 1 to 128 characters.

Request

Table 17-133 Request parameters

Parameter	Mandatory	Type	Description
plan_name	Yes	String	Name of a CU change. The name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_).
target_cu	Yes	Integer	Target value of the CU in the scheduled CU change.
start_hour	Yes	Integer	Specifies the start hour of the scheduled CU change.
start_minute	Yes	Integer	Specifies the start minute of a scheduled CU change.
repeat_day	Yes	Array of strings	Specifies the repetition period of a scheduled CU change. You can select one or more days from Monday to Sunday, or do not select any day. If this parameter is not specified, the scheduled CU change will be executed at the time specified by **start_hour: start_minute** after the current time. Example: `"repeat_day": ["MONDAY", "TUESDAY", "WEDNESDAY","SUNDAY"]`
valid_date_begin	No	Long	Start time of the validity period (13-digit timestamp)

Parameter	Mandatory	Type	Description
valid_date_end	No	Long	End time of the validity period (13-digit timestamp)
activate	No	Boolean	Indicates whether the scheduled CU change is activated. The default value is **true**, indicating that the change is activated.

Response

Table 17-134 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Example Request

Create a scaling plan named **plan_A** for the queue. According to the plan, the queue is scaled out at 20:30 on Monday, Tuesday, Wednesday, and Sunday, to 64 CUs.

```
{
    "plan_name": "plan_A",
    "target_cu": 64,
    "start_hour": 20,
    "start_minute": 30,
    "repeat_day": [
        "MONDAY",
        "TUESDAY",
        "WEDNESDAY",
        "SUNDAY"
    ],
    "valid_date_begin": 1590949800000,
    "valid_date_end": 1591727400000,
    "activate": true
}
```

Example Response

```
{
    "is_success": true,
    "message": ""
}
```


Status Codes

Table 17-135 describes status codes.

Table 17-135 Status codes

Status Code	Description
200	Scheduled CU change created successfully.
400	Request failure.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 17-136 Error codes

Error Code	Error Message
DLI.0999	Queue plans create failed. The plan plan_A can not generate a scale plan, please check all time settings for the plan.

17.7.2 Viewing a Scheduled CU Change (Discarded)

Function

This API is used to query the scheduled CU changes and list the changes of a specified queue.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  GET /v1/{project_id}/queues/{queue_name}/plans
- Parameter description

Table 17-137 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
queue_name	Yes	String	Name of the queue for which the scheduled CU change is to be deleted. The name contains 1 to 128 characters. Use commas (,) to separate multiple queue names.

Request

None

Response

Table 17-138 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
plans	No	Array of objects	Scheduled scaling plan information. For details, see **Table 17-139**.

Table 17-139 plans parameters

Parameter	Mandatory	Type	Description
id	No	Long	ID of a scheduled CU change.
plan_name	No	String	Name of a CU change. The name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_).
target_cu	No	Integer	Target value of the CU in the scheduled CU change.
start_hour	No	Integer	Start hour of a queue scaling plan, in the 24-hour format.

Parameter	Mandatory	Type	Description
start_minute	No	Integer	Specifies the start minute of a scheduled CU change.
repeat_day	Yes	Array of strings	Specifies the repetition period of a scheduled CU change. You can select one or more days from Monday to Sunday, or do not select any day. If this parameter is not specified, the scheduled CU change will be executed at the time specified by **start_hour: start_minute** after the current time. Example: `"repeat_day": ["MONDAY", "TUESDAY", "WEDNESDAY","SUNDAY"]`
valid_date_begin	No	Long	Start time of the validity period (13-digit timestamp)
valid_date_end	No	Long	End time of the validity period (13-digit timestamp)
activate	No	Boolean	Indicates whether the scheduled CU change is activated. The default value is **true**, indicating that the change is activated.
last_execute_time	No	Long	Time when the scaling plan was last executed.

Example Request

None

Example Response

```
{
    "is_success": true,
    "message": "",
    "plans": [
        {
            "id": 1,
            "plan_name": "plan_Aa",
            "target_cu": 32,
            "start_hour": 11,
            "start_minute": 15,
            "repeat_day": [
                "MONDAY",
                "TUESDAY",
                "WEDNESDAY",
                "SUNDAY"
            ],
            "activate": true,
            "last_execute_time": 1593573428857
        },
        {
            "id": 6,
            "plan_name": "plan_Ab",
            "target_cu": 16,
```



```
            "start_hour": 14,
            "start_minute": 25,
            "repeat_day": [
                "MONDAY",
                "TUESDAY",
                "WEDNESDAY",
                "SUNDAY",
                "THURSDAY",
                "FRIDAY",
                "SATURDAY"
            ],
            "activate": true,
            "last_execute_time": 1593584829260
        }
    ]
}
```


Status Codes

Table 17-140 describes status codes.

Table 17-140 Status codes

Status Code	Description
200	The query is successful.
400	Request failure.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 17-141 Error codes

Error Code	Error Message
DLI.0008	There is no queue named queue1.

17.7.3 Deleting Scheduled CU Changes in Batches (Discarded)

Function

This API is used to delete scheduled CU changes in batches.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  POST /v1/{project_id}/queues/{queue_name}/plans/batch-delete

- Parameter description

Table 17-142 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
queue_name	Yes	String	Name of the queue for which the scheduled CU change is to be deleted. The name contains 1 to 128 characters. Use commas (,) to separate multiple queue names.

Request

Table 17-143 Request parameters

Parameter	Mandatory	Type	Description
plan_ids	Yes	Array of Long	Scaling policy IDs of the queues you want to delete. For details, see **Viewing a Scheduled CU Change (Discarded)**. Example: "plan_ids": [8,10]

Response

Table 17-144 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Example Request

Delete the scaling plans whose IDs are 3 and 4.

```
{
  "plan_ids": [3,4]
}
```

Example Response

```
{
  "is_success": true,
  "message": ""
}
```

Status Codes

Table 17-145 describes status codes.

Table 17-145 Status codes

Status Code	Description
200	Deletion succeeded.
400	Request failure.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 17-146 Error codes

Error Code	Error Message
DLI.0002	The plans with id 8, 9 do not exist.

17.7.4 Deleting a Scheduled CU Change (Discarded)

Function

This API is used to delete a scheduled CU change for a queue with a specified ID.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  DELETE /v1/{project_id}/queues/{queue_name}/plans/{plan_id}
- Parameter description

Table 17-147 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
queue_name	Yes	String	Name of the queue for which the scheduled CU change is to be deleted. The name contains 1 to 128 characters. Use commas (,) to separate multiple queue names.
plan_id	Yes	Long	ID of scheduled CU change to be deleted. For details about how to obtain the IDs, see **Viewing a Scheduled CU Change (Discarded)**.

Request

None

Response

Table 17-148 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.

Example Request

None

Example Response

```
{
  "is_success": true,
```



```
  "message": ""
}
```

Status Codes

Table 17-149 describes status codes.

Table 17-149 Status codes

Status Code	Description
200	The directory is successfully deleted.
400	Request failure.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 17-150 Error codes

Error Code	Error Message
DLI.0002	The plan with id 8 does not exist.

17.7.5 Modifying a Scheduled CU Change (Discarded)

Function

This API is used to modify a scheduled CU change for a queue with a specified ID.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  PUT /v1/{project_id}/queues/{queue_name}/plans/{plan_id}
- Parameter description

Table 17-151 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
queue_name	Yes	String	Name of the queue for which the scheduled CU change is to be modified. The name contains 1 to 128 characters. Use commas (,) to separate multiple queue names.
plan_id	Yes	String	ID of scheduled CU change to be modified. Use commas (,) to separate multiple IDs. For details about how to obtain the IDs, see **Viewing a Scheduled CU Change (Discarded)**.

Request

Table 17-152 Request parameters

Parameter	Mandatory	Type	Description
plan_name	Yes	String	Name of a CU change. The name can contain only digits, letters, and underscores (_), but cannot contain only digits or start with an underscore (_).
target_cu	Yes	Integer	Target value of the CU in the scheduled CU change.
start_hour	Yes	Integer	Specifies the start hour of the scheduled CU change.
start_minute	Yes	Integer	Specifies the start minute of a scheduled CU change.
repeat_day	Yes	Array of strings	Specifies the repetition period of a scheduled CU change. You can select one or more days from Monday to Sunday, or do not select any day. If this parameter is not specified, the scheduled CU change will be executed at the time specified by **start_hour: start_minute** after the current time. Example: `"repeat_day": ["MONDAY", "TUESDAY", "WEDNESDAY","SUNDAY"]`
valid_date_begin	No	Long	Start time of the validity period (13-digit timestamp)

Parameter	Mandatory	Type	Description
valid_date_end	No	Long	End time of the validity period (13-digit timestamp)
activate	No	Boolean	Indicates whether the scheduled CU change is activated. The default value is **true**, indicating that the change is activated.

Response

Table 17-153 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	System prompt. If execution succeeds, the parameter setting may be left blank.
queue_name	No	String	Name of the queue for which the scheduled CU change is to be modified. The name contains 1 to 128 characters. Use commas (,) to separate multiple queue names.
plan_id	No	Integer	ID of scheduled CU change to be modified. Use commas (,) to separate multiple IDs.

Example Request

Modify the scaling plan named **plan_A**. After the modification, the number of CUs of the queue is scaled to 64 at 19:30 on Thursday and Friday, and the scaling plan is not activated.

```
{
    "plan_name": "plan_A",
    "target_cu": 64,
    "start_hour": 19,
    "start_minute": 30,
    "repeat_day": ["THURSDAY","FRIDAY"],
    "activate": false
}
```

Example Response

```
{
    "is_success": true,
    "message": "",
    "queue_name": "queue1",
```



```
  "plan_id": 3
}
```

Status Codes

Table 17-154 describes status codes.

Table 17-154 Status codes

Status Code	Description
200	The modification operations are successful.
400	Request failure.
500	Internal service error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

Table 17-155 Error codes

Error Code	Error Message
DLI.0999	Queue plans create failed. The plan plan_A can not generate a scale plan, please check all time settings for the plan.

17.8 Datasource Authentication-related APIs (Discarded)

17.8.1 Creating Datasource Authentication (Discarded)

Function

This API is used to create datasource authentication.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  POST /v2.0/{project_id}/datasource/auth-infos

- Parameter description

Table 17-156 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-157 Parameters

Parameter	Mandatory	Type	Description
auth_info_name	Yes	String	Authentication information name, which is unique in a project.
user_name	No	String	Username
password	No	String	Password
certificate_location	No	String	Path of the security cluster certificate. Currently, only OBS paths and CER files are supported.
datasource_type	Yes	String	Data source type. Available values are **CSS**, **KRB**, **passwd**, and **Kafka_SSL**.
krb5_conf	No	String	OBS path of the **krb5** configuration file
keytab	No	String	OBS path of the **keytab** configuration file
truststore_location	No	String	OBS path of the **truststore** configuration file
truststore_password	No	String	Password of the **truststore** configuration file
keystore_location	No	String	OBS path of the **keystore** configuration file
keystore_password	No	String	Password of the **keystore** configuration file
key_password	No	String	Key password

Response

Table 17-158 Parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	String	System prompt. If execution succeeds, the message may be left blank.

Example Request

Create a CSS datasource authentication.

```
{
    "auth_info_name": "test",
    "user_name": "admin",
    "password": "***",
    "certificate_location": "obs://test/CloudSearchService.cer",
    "datasource_type": "CSS"
}
```

Example Response

```
{
  "is_success": true,
  "message": "Upload success."
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

17.8.2 Listing Datasource Authentication Information (Discarded)

Function

This API is used to list the datasource authentication information in a project.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  GET /v2.0/{project_id}/datasource/auth-infos
- Parameter description

Table 17-159 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Table 17-160 query parameters

Parameter	Mandatory	Type	Description
limit	No	String	The maximum number of connections to be queried. The default value is **100**.
offset	No	String	The offset of the query result. The default value is **0**. Note that the connections are sorted by creation time.
auth_info_name	Yes	String	Authentication information name, which is unique in a project.

Request

None

Response

Table 17-161 Parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	String	System prompt. If execution succeeds, the message may be left blank.
count	int	Number of certificates

Parameter	Type	Description
auth_infos	List	Authentication information list. For details, see **Table 17-162**.

Table 17-162 auth_infos parameters

Parameter	Type	Description
auth_info_name	String	Username for logging in to the security cluster
user_name	String	Password for logging in to the security cluster
certificate_location	String	OBS path of the certificate specified during authentication creation
datasource_type	String	Data source type
create_time	Int	Timestamp when the authentication is created
update_time	Int	Timestamp when the authentication is updated
krb5_conf	String	OBS path of the **krb5** configuration file
keytab	String	OBS path of the **keytab** configuration file
truststore_location	String	OBS path of the **truststore** configuration file
keystore_location	String	OBS path of the **keystore** configuration file
owner	String	Username

Example Request

None

Example Response

```
{
    "count": 19,
    "auth_infos": [{
        "auth_info_name": "lan2",
        "datasource_type": "Kafka_SSL",
        "create_time": 1578896427789,
        "update_time": 1578898059677,
        "owner": "ei_dlics_d00352221",
        "truststore_location": "obs://lan-1/cer/truststore.jks",
```



```
        "keystore_location": "obs://lan-1/cer/keystore.jks"
    }, {
        "auth_info_name": "lan3",
        "datasource_type": "Kafka_SSL",
        "create_time": 1578899029971,
        "update_time": 1578901678150,
        "owner": "ei_dlics_d00352221",
        "truststore_location": "obs://lan-1/cer/truststore1.jks",
        "keystore_location": "obs://lan-1/cer/keystore.jks"
    }]
}
```


Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

17.8.3 Updating Datasource Authentication (Discarded)

Function

This API is used to update authentication information.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  PUT /v2.0/{project_id}/datasource/auth-infos

- Parameter description

  Table 17-163 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Parameter	Mandatory	Type	Description
auth_info_name	Yes	String	Name of the updated datasource authentication.
user_name	No	String	New username for logging in to the security cluster
password	No	String	New password for logging in to the security cluster
krb5_conf	No	String	OBS path of the **krb5** configuration file
keytab	No	String	OBS path of the **keytab** configuration file
truststore_location	No	String	OBS path of the **truststore** configuration file
truststore_password	No	String	Password of the **truststore** configuration file
keystore_location	No	String	OBS path of the **keystore** configuration file
keystore_password	No	String	Password of the **keystore** configuration file

Response

Table 17-164 Parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	String	None

Example Request

Change the name of datasource authentication to **rds3**.

```
{
    "auth_info_name": "rds3"
}
```

Example Response

```
{
  "is_success": true,
```



```
 "message": "Update success."
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

17.8.4 Deleting Datasource Authentication (Discarded)

Function

This API is used to delete datasource authentication.

NOTE

This API has been discarded and is not recommended.

URI

- URI format
  DELETE /v2.0/{project_id}/datasource/auth-infos/{auth_info_name}
- Parameter description

Table 17-165 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
auth_info_name	Yes	String	Name of the authentication information you want to delete.

Response

Table 17-166 Parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	String	None

Example Request

None

Example Response

```
{
    "is_success": true,
    "message": "Deleted success."
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

17.9 APIs Related to Enhanced Datasource Connections (Discarded)

17.9.1 Creating a Route (Discarded)

Function

This API is used to create a datasource connection route.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  POST /v2.0/{project_id}/datasource/enhanced-connections/{connection_id}/routes

- Parameter description

  Table 17-167 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
connection_id	Yes	String	Datasource connection ID

Request

Table 17-168 Parameters

Parameter	Mandatory	Type	Description
name	Yes	String	Route name. The value can contain 1 to 64 characters.
cidr	Yes	String	Route network range

Response

Table 17-169 Parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	String	System prompt. If execution succeeds, the message may be left blank.

Example Request

Create a route. Set the next-hop address of the enhanced datasource connection to **127.0.0.0**.


```
{
  "name": "route",
  "cidr": "127.0.0.0"
}
```

Example Response

```
{
  "is_success": true,
  "message": ""
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

17.9.2 Deleting a Route (Discarded)

Function

This API is used to delete a datasource connection route.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  DELETE /v2.0/{project_id}/datasource/enhanced-connections/{connection_id}/routes/{name}

- Parameter description

  Table 17-170 URI parameters

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.
connection_id	Yes	String	Datasource connection ID

Parameter	Mandatory	Type	Description
name	Yes	String	Route name

Request

None

Response

Table 17-171 Parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	String	System prompt. If execution succeeds, the message may be left blank.

Example Request

None

Example Response

```
{
  "is_success": true,
  "message": "Deleted"
}
```

Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

17.10 Template-related APIs (Discarded)

17.10.1 Querying All Sample SQL Templates (Discarded)

Function

This API is used to query all sample SQL templates.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  GET /v1.0/{project_id}/sqls/sample
- Parameter description

Table 17-172 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

None

Response

Table 17-173 Parameters

Parameter	Type	Description
is_success	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	String	System prompt. If execution succeeds, the message may be left blank. If the execution fails, the value will be the cause of the failure.
sqls	Array of Object	Sample template information. For details, see **Table 17-174**.
sqlCount	Integer	Number of sample templates.

Table 17-174 sqls parameters

Parameter	Type	Description
lang	String	Language
name	String	Name of a sample template
sql	String	Template content
description	String	Template description
group	String	Template group

Example Request

None

Example Response


```
{
  "is_success": true,
  "message": "",
  "sqlCount": 1,
  "sqls": [
    {
      "lang": "zh",
      "name": "Q1_ Price Summary Report",
      "sql": "-- Q1: Query pricing reports.\r\n-- Query statistics on paid and delivered offerings in a specified period in a single table lineitem, including the charging, delivery, discount, tax, and average price.\r\n-- Feature: single-table query with grouping, sorting, and aggregate operations. This query read 95% to 97% rows of data in the table.\r\nSELECT\r\n  l_returnflag,\r\n  l_linestatus,\r\n  sum(l_quantity) AS sum_qty,\r\n  sum(l_extendedprice) AS sum_base_price,\r\n  sum(l_extendedprice * (1 - l_discount)) AS sum_disc_price,\r\n  sum(l_extendedprice * (1 - l_discount) * (1 + l_tax)) AS sum_charge,\r\n  avg(l_quantity) AS avg_qty,\r\n  avg(l_extendedprice) AS avg_price,\r\n  avg(l_discount) AS avg_disc,\r\n  count(*) AS count_order\r\nFROM\r\n  tpch.lineitem\r\nWHERE\r\n  l_shipdate <= DATE \"1998-12-01\" - INTERVAL \"90\" DAY\r\nGROUP BY\r\n  l_returnflag,\r\n  l_linestatus\r\nORDER BY\r\n  l_returnflag,\r\n  l_linestatus;",
      "description": " Query the lineitem of a table for statistics on paid and delivered offerings in a specified period, including the charging, delivery, discount, tax, and average price information.",
      "group": "tpchQuery"
    }
  ]
}
```


Status Codes

Status Code	Description
200	OK

Error Codes

For details, see **Error Codes**.

17.11 APIs Related to Flink Jobs (Discarded)

17.11.1 Querying Job Monitoring Information (Discarded)

Function

This API is used to query job monitoring information. You can query monitoring information about multiple jobs at the same time.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  POST /v1.0/{project_id}/streaming/jobs/metrics
- Parameter description

Table 17-175 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-176 Request parameter

Parameter	Mandatory	Type	Description
job_ids	Yes	Array of Long	List of job IDs.

Response

Table 17-177 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Indicates whether the request is successful.
message	No	String	Message content.
metrics	No	Object	Information about a job list. For details, see **Table 17-178**.

Table 17-178 payload parameters

Parameter	Mandatory	Type	Description
jobs	No	Array of objects	Monitoring information about all jobs. For details, see **Table 17-179**.

Table 17-179 jobs parameters

Parameter	Mandatory	Type	Description
job_id	No	Long	Job ID.
metrics	No	Object	All input and output monitoring information about a job. For details, see **Table 17-180**.

Table 17-180 metrics parameters

Parameter	Mandatory	Type	Description
sources	No	Array of objects	All source streams. For details, see **Table 17-181**.
sinks	No	Array of objects	All sink streams. For details, see **Table 17-181**.
total_read_rate	No	Double	Total read rate.
total_write_rate	No	Double	Total write rate.

Table 17-181 source/sinks parameters

Parameter	Mandatory	Type	Description
name	No	String	Name of the source or sink stream.
records	No	Long	Total number of records.
corrupted_records	No	Long	Number of dirty data records.

Example

- Example request

```
{
   "job_ids": [298765, 298766]
}
```

- Example response

```
{
   "is_success": true,
   "message": "Message content",
   "metrics": {
      "jobs": [
         {
            "job_id": 0,
            "metrics": {
               "sources": [
                  {
                     "name": "Source: KafKa_6070_KAFKA_SOURCE",
                     "records": 0,
                     "corrupted_records": 0
                  }
               ],
               "sinks": [
                  {
                     "name": "Source: KafKa_6070_KAFKA_SOURCE",
                     "records": 0,
                     "corrupted_records": 0
                  }
               ],
               "total_read_rate": 100,
               "total_write_rate": 100
            }
         }
      ]
   }
}
```

Status Codes

Table 17-182 Status codes

Status Code	Description
200	The query of job monitoring information succeeds.
400	The input parameter is invalid.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

17.11.2 Granting OBS Permissions to DLI

Function

This API is used to grant DLI the permission to access OBS buckets for saving job checkpoints and run logs.

NOTE

This API has been discarded and is not recommended.

URI

- URI format

  POST /v1.0/{project_id}/dli/obs-authorize
- Parameter description

Table 17-183 URI parameter

Parameter	Mandatory	Type	Description
project_id	Yes	String	Project ID, which is used for resource isolation. For details about how to obtain its value, see **Obtaining a Project ID**.

Request

Table 17-184 Request parameter

Parameter	Mandatory	Type	Description
obs_buckets	Yes	Array of Strings	List of OBS buckets.

Response

Table 17-185 Response parameters

Parameter	Mandatory	Type	Description
is_success	No	Boolean	Whether the request is successfully executed. Value **true** indicates that the request is successfully executed.
message	No	String	Message content.

Example Request

Grant DLI the permission to access the OBS bucket **bucket1** so that DLI can save job checkpoints and run logs to the bucket.

```
{
   "obs_buckets": [
      "bucket1"
   ]
}
```

Example Response

```
{
   "is_success": "true",
   "message": "The following OBS bucket is authorized successfully, bucket1."
}
```

Status Codes

Table 17-186 describes the status code.

Table 17-186 Status codes

Status Code	Description
200	Authorization succeeds.
400	Request error.

Error Codes

If an error occurs when this API is invoked, the system does not return the result similar to the preceding example, but returns the error code and error information. For details, see **Error Codes**.

18 Public Parameters

18.1 Status Codes

Table 18-1 describes status codes.

Table 18-1 Status codes

Status Code	Message	Description
100	Continue	The client should continue with its request. This interim response is used to inform the client that part of the request has been received and has not yet been rejected by the server.
101	Switching Protocols	The protocol should be switched. The protocol can only be switched to a newer protocol. For example, the current HTTP protocol is switched to a later version of HTTP.
200	Success	The request has been fulfilled. This indicates that the server has provided the requested web page.
201	Created	The request is successful and the server has created a new resource.
202	Accepted	The request has been accepted, but the processing has not been completed.
203	Non-Authoritative Information	Unauthorized information. The request is successful.

Status Code	Message	Description
204	NoContent	The server has successfully processed the request, but does not return any content. The status code is returned in response to an HTTP OPTIONS request.
205	Reset Content	The server has successfully processed the request, but does not return any content.
206	Partial Content	The server has successfully processed the partial GET request.
300	Multiple Choices	There are multiple options for the location of the requested resource. The response contains a list of resource characteristics and addresses from which a user terminal (such as a browser) can choose the most appropriate one.
301	Moved Permanently	The requested resource has been assigned a new permanent URI, and the new URI is contained in the response.
302	Found	The requested resource resides temporarily under a different URI.
303	See Other	The response to the request can be found under a different URI, and should be retrieved using a GET or POST method.
304	Not Modified	The requested resource has not been modified. In such a case, there is no need to retransmit the resource since the client still has a previously-downloaded copy.
305	Use Proxy	The requested resource is available only through a proxy.
306	Unused	The HTTP status code is no longer used.
400	BadRequest	Invalid request. The client should not repeat the request without modifications.
401	Unauthorized	This status code is returned after the client provides the authentication information, indicating that the authentication information is incorrect or invalid.
402	Payment Required	This status code is reserved for future use.

Status Code	Message	Description
403	Forbidden	The server has received the request and understood it, but the server is refusing to respond to it. The client should modify the request instead of re-initiating it.
404	NotFound	The requested resource cannot be found. The client should not repeat the request without modifications.
405	MethodNotAllowed	A request method is not supported for the requested resource. The client should not repeat the request without modifications.
406	Not Acceptable	The server could not fulfill the request according to the content characteristics of the request.
407	Proxy Authentication Required	This code is similar to 401, but indicates that the client must first authenticate itself with the proxy.
408	Request Time-out	The server has timed out waiting for the request. The client may repeat the request without modifications at any time later.
409	Conflict	The request could not be processed due to a conflict in the request. This status code indicates that the resource that the client is attempting to create already exists, or that the request has failed to be processed because of the update of the conflict request.
410	Gone	The requested resource cannot be found. The status code indicates that the requested resource has been deleted permanently.
411	Length Required	The server is refusing to process the request without a defined **Content-Length**.
412	Precondition Failed	The server does not meet one of the preconditions that the requester puts on the request.

Status Code	Message	Description
413	Request Entity Too Large	The server is refusing to process a request because the request entity is too large for the server to process. The server may disable the connection to prevent the client from sending requests consecutively. If the server is only temporarily unable to process the request, the response will contain a **Retry-After** header field.
414	Request-URI Too Large	The Request-URI is too long for the server to process.
415	Unsupported Media Type	The server does not support the media type in the request.
416	Requested range not satisfiable	The requested range is invalid.
417	Expectation Failed	The server has failed to meet the requirements of the **Expect** request-header field.
422	UnprocessableEntity	The request was well-formed but was unable to be followed due to semantic errors.
429	TooManyRequests	The client sends excessive requests to the server within a given time (exceeding the limit on the access frequency of the client), or the server receives excessive requests within a given time (beyond its processing capability). In this case, the client should resend the request after the time specified in the **Retry-After** header of the response has elapsed.
500	InternalServerError	The server is able to receive the request but unable to understand it.
501	Not Implemented	The server does not support the requested function.
502	Bad Gateway	The server was acting as a gateway or proxy and received an invalid request from the remote server.
503	ServiceUnavailable	The requested service is invalid. The client should not repeat the request without modifications.
504	ServerTimeout	The request cannot be fulfilled within a given time. This status code is returned to the client only when the **Timeout** parameter is specified in the request.

Status Code	Message	Description
505	HTTP Version not supported	The server does not support the HTTP protocol version used in the request.

18.2 Error Codes

If an error occurs in API calling, no result is returned. Identify the cause of error based on the error codes of each API. If an error occurs in API calling, HTTP status code 4*xx* or 5*xx* is returned. The response body contains the specific error code and information. If you are unable to identify the cause of an error, contact technical personnel and provide the error code so that we can help you solve the problem as soon as possible.

Format of an Error Response Body

If an error occurs during API calling, the system returns an error code and a message to you. The following shows the format of an error response body:

```
{
    "error_msg": "The format of message is error",
    "error_code": "DLI.0001"
}
```

In the preceding information, **error_code** is an error code, and **error_msg** describes the error.

Table 18-2 Exceptions

Parameter	Parameter Type	Description
error_code	String	Error code. For details, see **Table 18-3**.
error_msg	String	Error details.

Error Code Description

If an error code starting with **APIGW** is returned after you call an API, you can rectify the fault by referring to the instructions provided in "API Gateway Error Codes".

Table 18-3 Error codes

Status Code	Error Code	Error Message
400	DLI.0001	Parameter check errors occur.
400	DLI.0002	The object does not exist.

Status Code	Error Code	Error Message
400	DLI.0003	SQL permission verification fails.
400	DLI.0004	SQL syntax parsing errors occur.
400	DLI.0005	SQL semantics parsing errors occur.
400	DLI.0006	The object exists.
400	DLI.0007	The operation is not supported.
400	DLI.0008	Metadata errors occur.
400	DLI.0009	System restrictions.
400	DLI.0011	The file permission check fails.
400	DLI.0012	Resource objects are unavailable.
401	DLI.0013	User authentication errors occur.
401	DLI.0014	Service authentication errors occur.
400	DLI.0015	Token parsing error.
400	DLI.0016	The identity and role are incorrect.
400	DLI.0018	Data conversion errors occur.
400	DLI.0019	The task times out.
400	DLI.0100	The result expires.
404	DLI.0023	No related resources were found.
400	DLI.0999	Server-side errors occur.
400	DLI.1028	The quota is insufficient.

Example

If no queue named **testqueue** exists, the following error message is displayed when you submit a job submission request:

```
{
  "error_code": "DLI.0002",
  "error_msg": "There is no queue named testqueue"
}
```

18.3 Obtaining a Project ID

Scenario

A project ID is required for some URLs when an API is called. Obtain a project ID using either of the following methods:

- **Obtain the Project ID by Calling an API**
- **Obtain the Project ID from the Console**

Obtaining a Project ID by Calling an API

You can obtain a project ID by calling an API.

The API for obtaining a project ID is **GET https://{Endpoint}/v3/projects**. **{Endpoint}** indicates the endpoint of IAM, which can be obtained from the administrator.

```
{
    "projects": [
        {
            "domain_id": "65382450e8f64ac0870cd180d14e684b",
            "is_domain": false,
            "parent_id": "65382450e8f64ac0870cd180d14e684b",
            "name": "project_name",
            "description": "",
            "links": {
                "next": null,
                "previous": null,
                "self": "https://www.example.com/v3/projects/a4a5d4098fb4474fa22cd05f897d6b99"
            },
            "id": "a4a5d4098fb4474fa22cd05f897d6b99",
            "enabled": true
        }
    ],
    "links": {
        "next": null,
        "previous": null,
        "self": "https://www.example.com/v3/projects"
    }
}
```

Obtaining a Project ID from the Management Console

To obtain a project ID from the Console, perform the following operations:

1. Log in to the management console.
2. Hover the mouse pointer over the username in the upper right corner and choose **My Credentials** from the drop-down list.

   On the **Projects** tab of the **API Credentials** page, view project IDs.

18.4 Obtaining an Account ID

An account ID (domain-id) is required for some URLs when an API is called. To obtain an account ID, perform the following operations:

1. Log in to the management console.
2. Hover the cursor on the username in the upper right corner and select **My Credentials** from the drop-down list.
3. On the **API Credentials** page, view **Account ID**.

