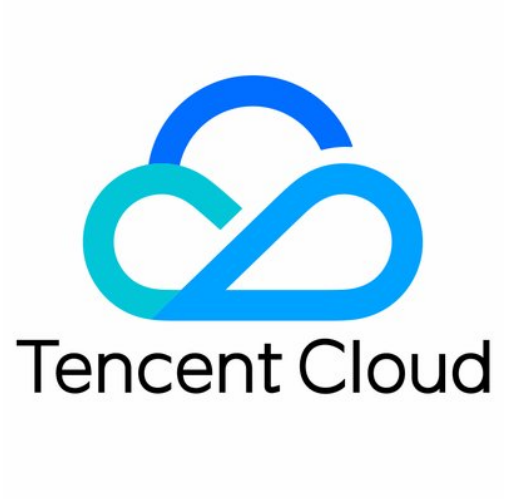
Data Lake Compute

Getting Started

Product Documentation



Contents

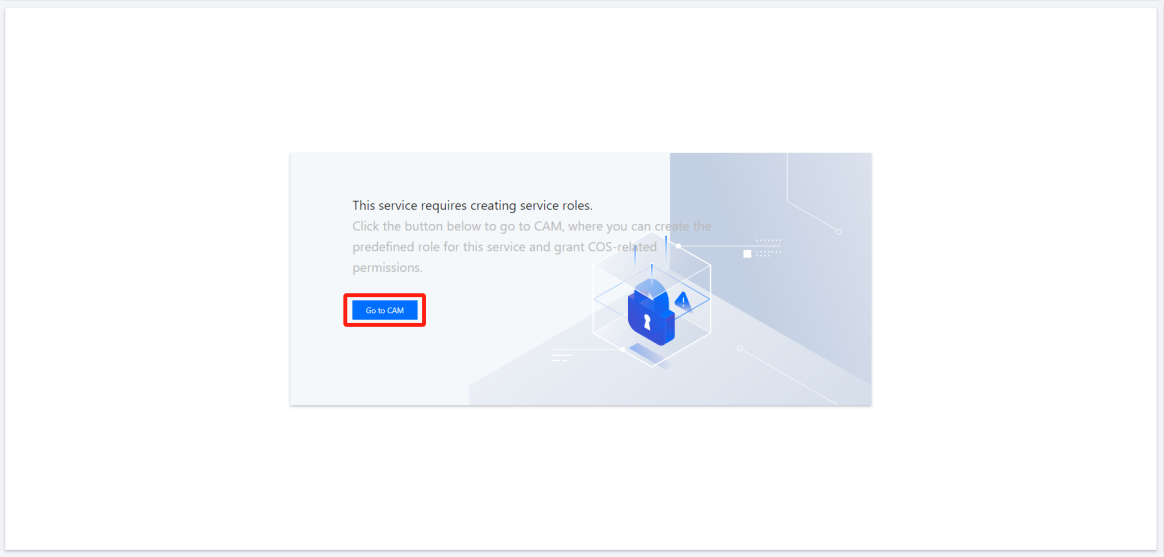
Getting Started
Complete Process for New User Activation

Last updated：2025-03-12 18:03:39

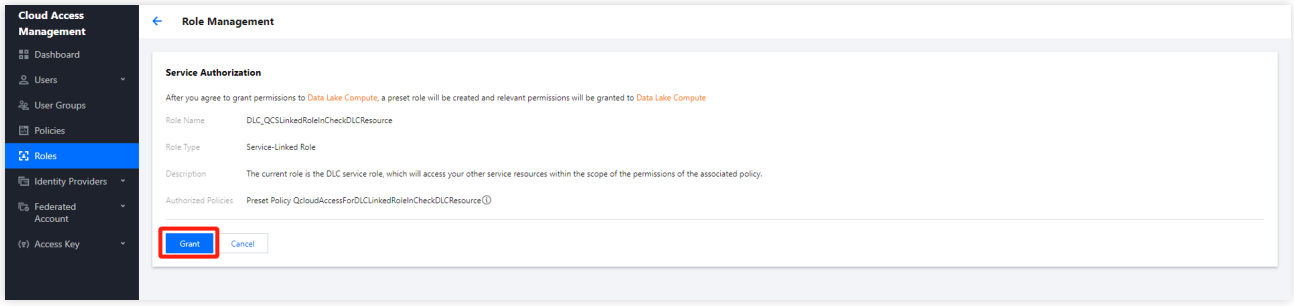
Before official and proper use of Data Lake Compute (DLC), you need to configure the initialization parameters and permissions in the target region in advance with the Tencent Cloud root account or DLC administrator account.
To avoid unnecessary errors, it is recommended that common users use DLC after the administrator completes the configurations.

Activating the DLC Service for the Root Account(Performed with the Tencent Cloud Root Account)

1. Log in to the DLC console.
2. Click **Go to CAM**.
3. Authorize the activation of DLC Data Lake Service.



4. In Role Management, click **Grant**.



Creating a Tencent Cloud Sub-account and Adding DLC Service Policies Under the Root Account(Performed with the Tencent Cloud Root Account)

To enable multiple accounts to use the DLC service collaboratively, activate the DLC service as follows:

Creating a Tencent Cloud Sub-account

To enable a sub-account to access DLC, go to the Create User page of CAM for configuration.

Adding the DLC Service Policy QcloudDLCFullAccess

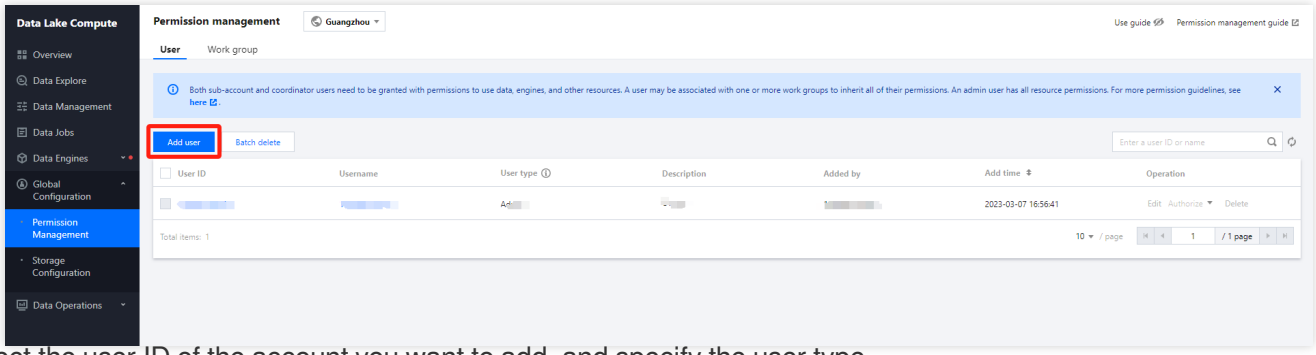
1. On the list of users in CAM page, select the user to be authorized.
2. Click **Authorize**.
3. In the pop-up window, enter DLC and select QcloudDLCFullAccess.

Adding the Sub-account as a DLC Account

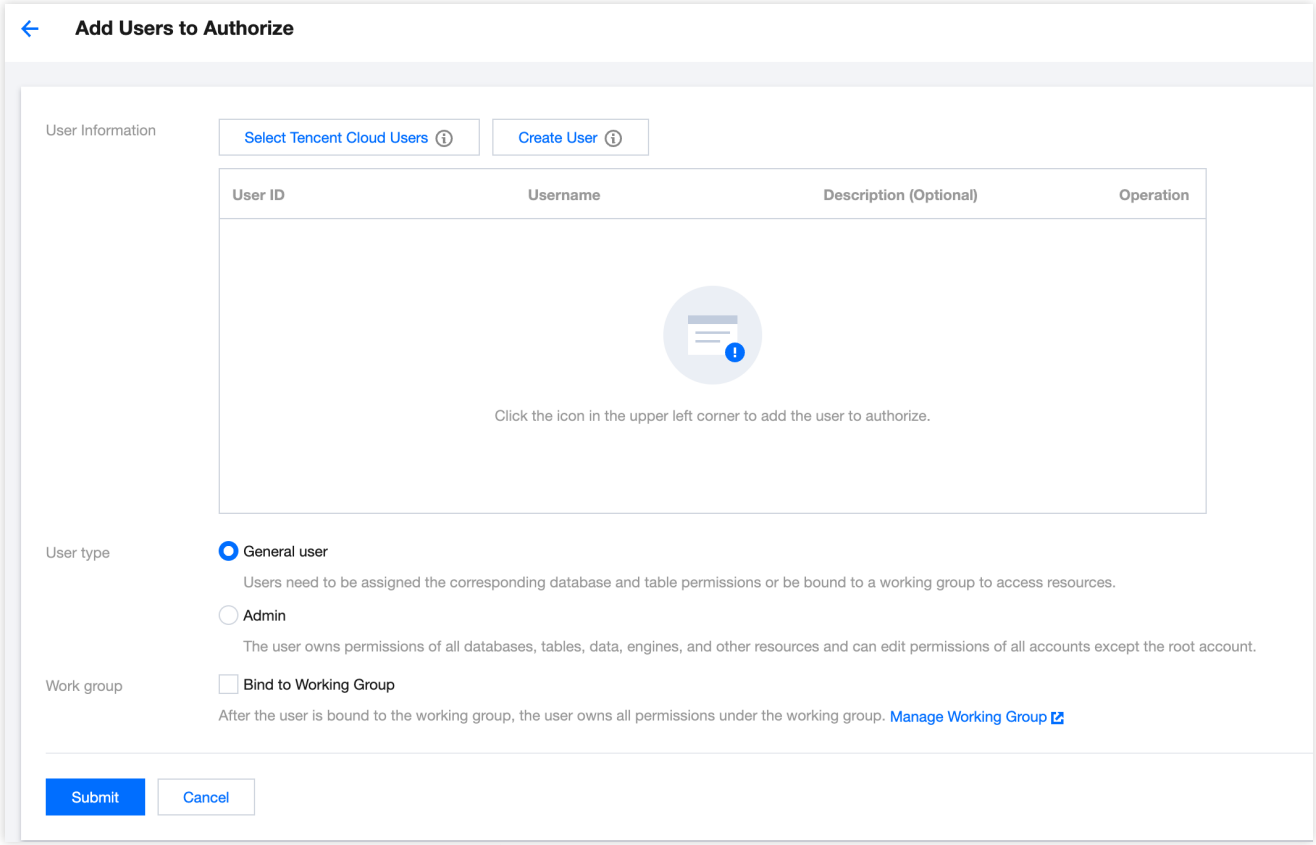
Note:

Operation permission: The first time adding a sub-account as a DLC account, use the Tencent Cloud root account.

1. Log in to the DLC console, select Permission Management, and click **Add user**.



2. Select the user ID of the account you want to add, and specify the user type.



DLC Account Types and Permission Scopes

Permission & Operation	DLC Administrator	Common User
Data permissions	All permissions	No permission by default, and permissions need to be authorized by a DLC administrator.
Standard engine	All permissions	No permission by default, and permissions need to be authorized by a DLC administrator.
User management	Available	Unavailable
Work group management	Available	Unavailable

Authorization scope	All permissions	**Authorizable** permissions

Causes of Addition Failures

If an error occurs when you add the account as a DLC account, check whether any of the following issues occur. If not, contact us for help.

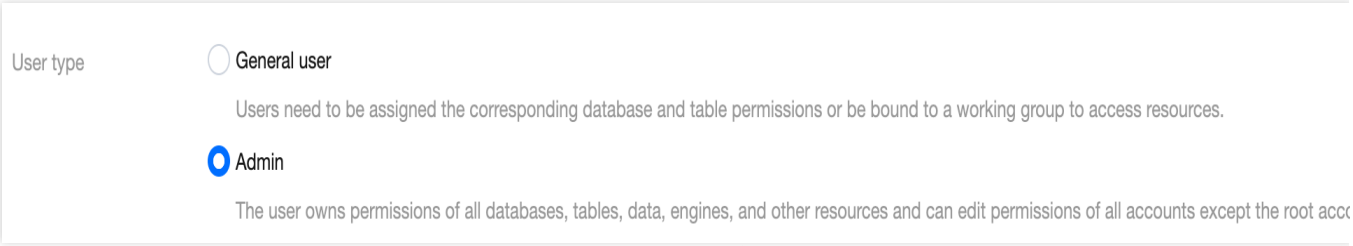
1. Repeated addition: The account has already been added as a DLC account.
2. Account input error: An incorrect account is entered.

Adding a DLC Administrator Account

Note:

Operation permission: The first time adding a sub-account as a DLC administrator account, use the Tencent Cloud root account. For subsequent addition of sub-accounts as DLC administrator accounts, you can use the added DLC administrator account.

The add method is consistent with the previous chapter (adding an account as a DLC user). When selecting the type of user, choose DLC administrator.



DLC Administrator Account Types and Management Scope

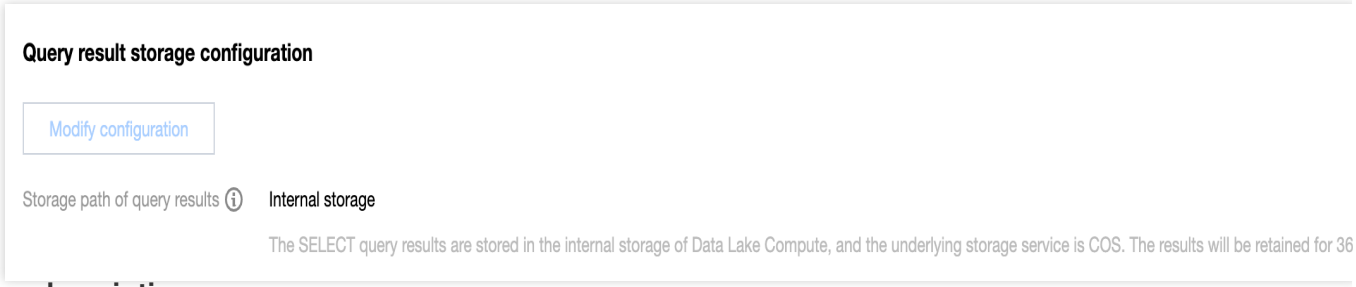
Administrator Type	Manageable Platform	Creating a Tencent Cloud Sub-account	Adding a DLC Policy	Adding a Tencent Cloud Sub-account as a DLC User	DLC Computing and Data Permissions
Root account (owner)	CAM/DLC	Allowed	Allowed	Allowed	Allowed
DLC administrator account	DLC	Disallowed	Disallowed	Allowed	Allowed

Configuring the Storage Path of Query Results

Note:

Operation permission: Configured with the root account or DLC administrator account.

Enter Storage Configuration in the DLC console and configure the storage path for requry results on the Overview page or the Storage Configuration page. After the configuration is completed, the query results are stored in the specified Cloud Object Storage (COS) path or the managed storage device of DLC.



Feature description:

1. DLC internal storage: The SELECT query results are stored in the DLC storage. The underlying storage is COS. The results can be stored for 36 hours.
2. User storage: The SELECT query results are stored in the bucket path on COS. You need to check whether the COS-related permissions are granted.
3. Metadata acceleration bucket: The performance of query and analysis in the local region can be improved.
4. For internal tables, the metadata acceleration bucket can be enabled directly. For external tables, you need to check whether the metadata acceleration bucket can be enabled based on the engine permission.

Note: A shared engine cannot be bound to a metadata acceleration bucket. When a user selects the user storage path, the exclusive engine needs to be bound to a metadata acceleration bucket before querying takes effect.

Purchasing an Engine

Note:

Operation permission: Purchased with the root account or an account having the financial permission.

You can purchase different types of engines based on your business requirements. Engines are classified into standard engines and SuperSQL engines. The two types of engines support different SQL syntaxes. The standard engine supports the native syntax and behavior, while the SuperSQL engine supports the DLC-developed SuperSQL syntax.

1. Purchase method: Go to the Standard Engine page.
2. Click **Create resource** to enter the purchase page.

Note:

1. **The engine is divided into the standard engine and SuperSQL engine. The difference between them is:** They support different SQL syntaxes. The standard engine supports native syntax and behaviors, while the SuperSQL

engine supports DLC's self-developed SuperSQL syntax.
2. **Engine Specification Purchase Advice**: Since a 16-CU cluster is relatively small in scale, it is advisable to use it only for testing scenarios. For real production scenarios, it is recommended to purchase a cluster with 64 CUs or more.

Standard Engine Permission Management

Note:
Operation permission: Configured with the root account or a sub-account having the CAM permissions.
Tencent Cloud CAM controls the DLC standard engine permissions. To ensure that the sub-account can use the DLC standard engine smoothly, you need to use the root account to authorize the sub-account the permissions. After the standard engine is created, all sub-accounts with the **QcloudDLCFullAccess policy (all read/write and access permissions in the DLC console)** have the permissions to use, manage, and monitor the standard engine. To achieve granular permission management of the standard engine, for example, user A only having the permission of engine A, you can create custom policies.

Scenario	Operation
Scenario 1: The sub-account has all standard engine permissions.	Associate the preset QcloudDLCFullAccess policy with the sub-account.
Scenario 2: The sub-account has partial standard engine permissions.	Create a custom policy.

Granting All Standard Engine Permissions to a Sub-account

After you log in to the DLC console by using the root account or a sub-account having CAM operation permissions, find the sub-account in the sub-account list, click Authorize in the **Operation** column, search for and select **QcloudDLCFullAccess**, and click **OK**.

Granting Partial Standard Engine Permissions to a Sub-account

DLC supports resource-level authentication based on CAM tags. You can use tags to manage the existing standard engine resources and engine-related API permissions of DLC by category, achieving multidimensional resource management by category and granular authorization. For details about Tencent Cloud tags, see Tag.
Based on Tencent Cloud tags, you can quickly achieve the following effects for DLC standard engine resources:
All users in department A can only use the standard engine resources associated with the tag for department A, and cannot use the standard engines associated with the tag for another department.
When users in department A create DLC standard engine resources, the resources should be associated with the tag for department A. If the resource is not associated with any tag or is associated with the tag for another department (rather than department A), the creation fails (optional).

Operation Steps

Step 1: Creating a Tag

1. Enter the Tag List page and click Create Tag.
2. Set **Tag Key** and **Tag Value**, click **OK**, and then a tag is created. For example, to create a tag with the department name Analyze, enter department for Tag Key and Analyze for Tag Value.

Step 2: Tagging the Standard Engine

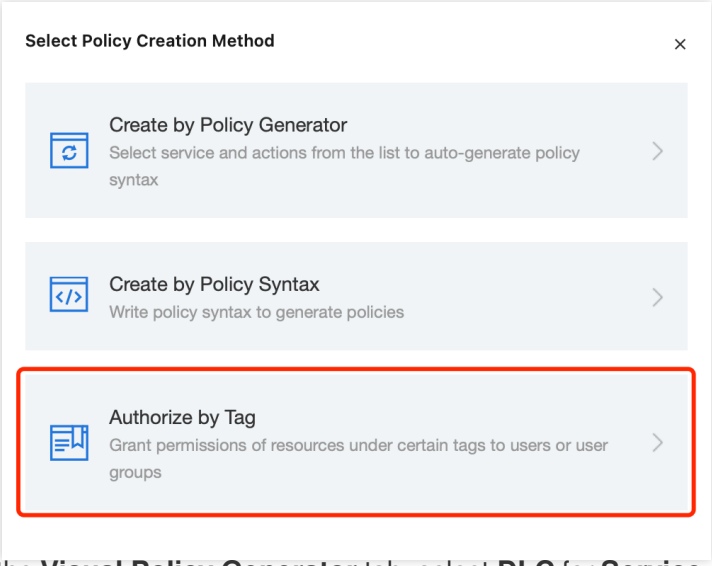
Log in to the DLC console and select Standard engine. In the tag option, select a tag you want to bind. For detailed operations on tagging a standard engine, see Associating Tag with Private Engine Resource.

Note:

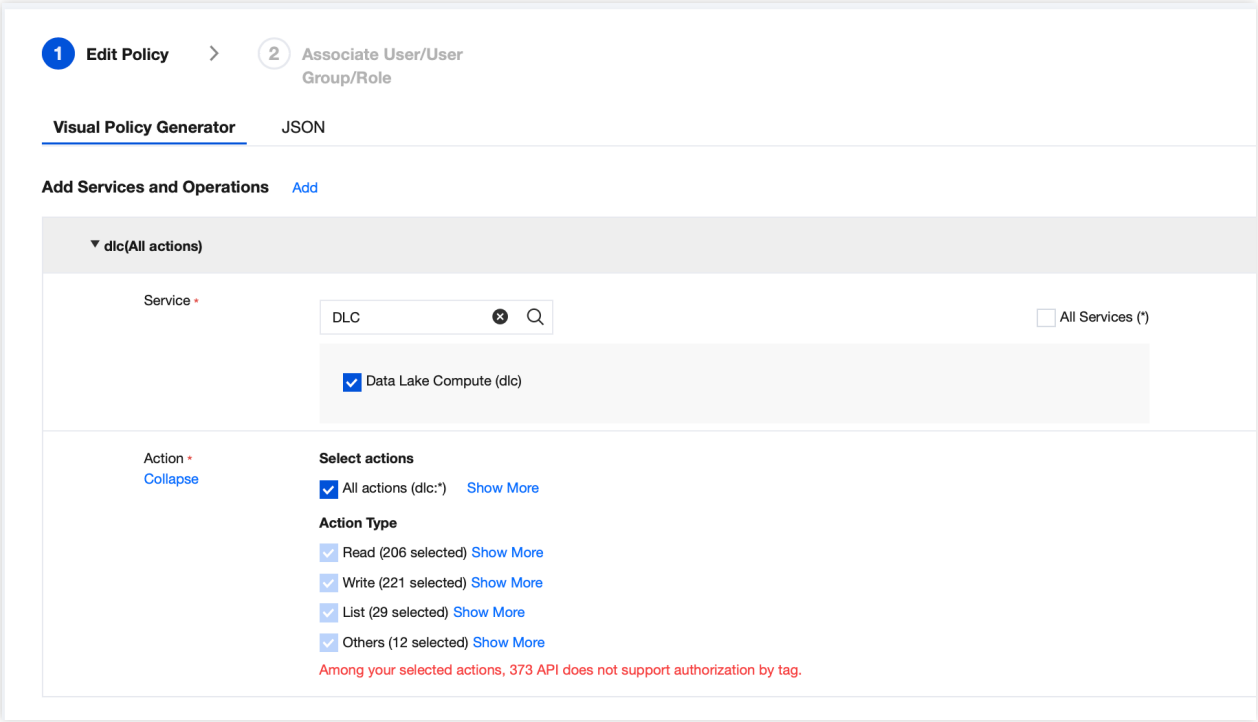
Once a specific tag is associated with a standard engine, such as “department: Analyze” in the above example, only the users who are associated with that tag can use and manage the standard engine.

Step 3: Creating a Custom Policy

1. Log in to the CAM console.
2. Click **Create Custom Policy**. In the pop-up window, select **Authorize by Tag**.



3. On the **Visual Policy Generator** tab, select **DLC** for **Service** and select **All actions (dlc:*) for Action.**



Note:

To restrict the permissions to terminate, create, or modify a standard engine, you can deselect the APIs under All actions. The APIs are described in the following table.

Situation	DLC API to be Deselected
Unable to terminate an engine	DeleteDataEngine
Unable to create an engine	CreateDataEngine
Unable to modify an engine	UpdateDataEngine

4. Select the “department: Analyze” tag created previously. By default, resource_tag under Select Condition Key is selected.

You can select request_tag,so that the users in the Analyze department will be required to associate the new DLC standard engine with the tag “department: Analyze” when creating an engine. For more introduction and usage of request_tag, see request_tag.

5. Click **Next** and CAM creates multiple split sub-policies. You can enter a name that is easy to search for a split sub-custom policy, such as “DLC-department-analyze-tag-policy”. Next to **Authorized Users** or **Authorized User Groups,** select the user/user group that needs to be associated with this custom policy. For example, associate all employee sub-accounts of the Analyze department in this example.

6. Click **Complete** and the custom policy is created. After the above custom policy is created, all DLC users associated with this custom policy can only access the standard engine tagged with “department: Analyze”.

Note:

1. To facilitate subsequent user maintenance, it is recommended that a user group be associated.

2. If a user associated with a custom policy has already been associated with the preset QcloudDLCFullAccess policy, the user still has all standard engine permissions.

SuperSQL Engine Permission Management

Note:

Operation permission: Configured by the root account or DLC admin.

Automatic Authorization Of Engine Operation Permissions (Only Supports SuperSQL Engine)

DLC supports enabling SuperSQL engine operation permissions by default. After enabling, all users will have the following permissions for this engine by default:

Usage: Use this engine for task execution.

Operation: Suspend or hang up the engine.

Monitoring: Monitor the usage and Ops of the engine.

Note:

1. After disabling, administrators will continue to have all engine permissions by default, while ordinary users require administrators to add permissions on the permission management page.

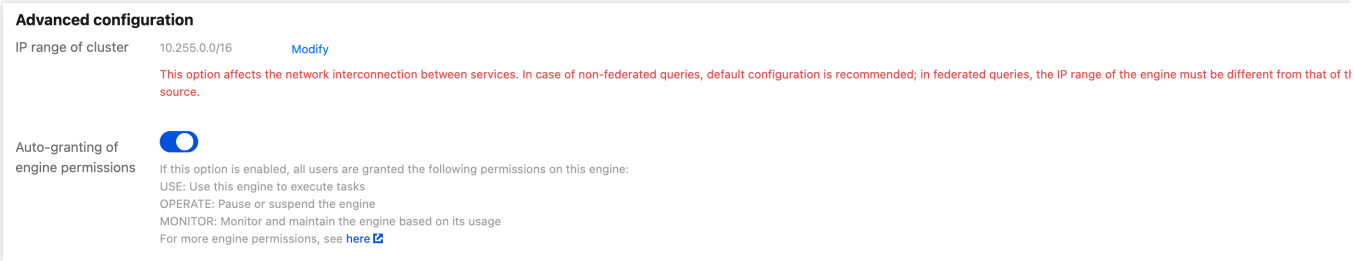
2. The original ordinary users' permissions are not affected and can be deleted by going toPermission Management page.

3. Subsequent newly created ordinary users will not have usage permissions and need to be manually added on the Permission Management page.

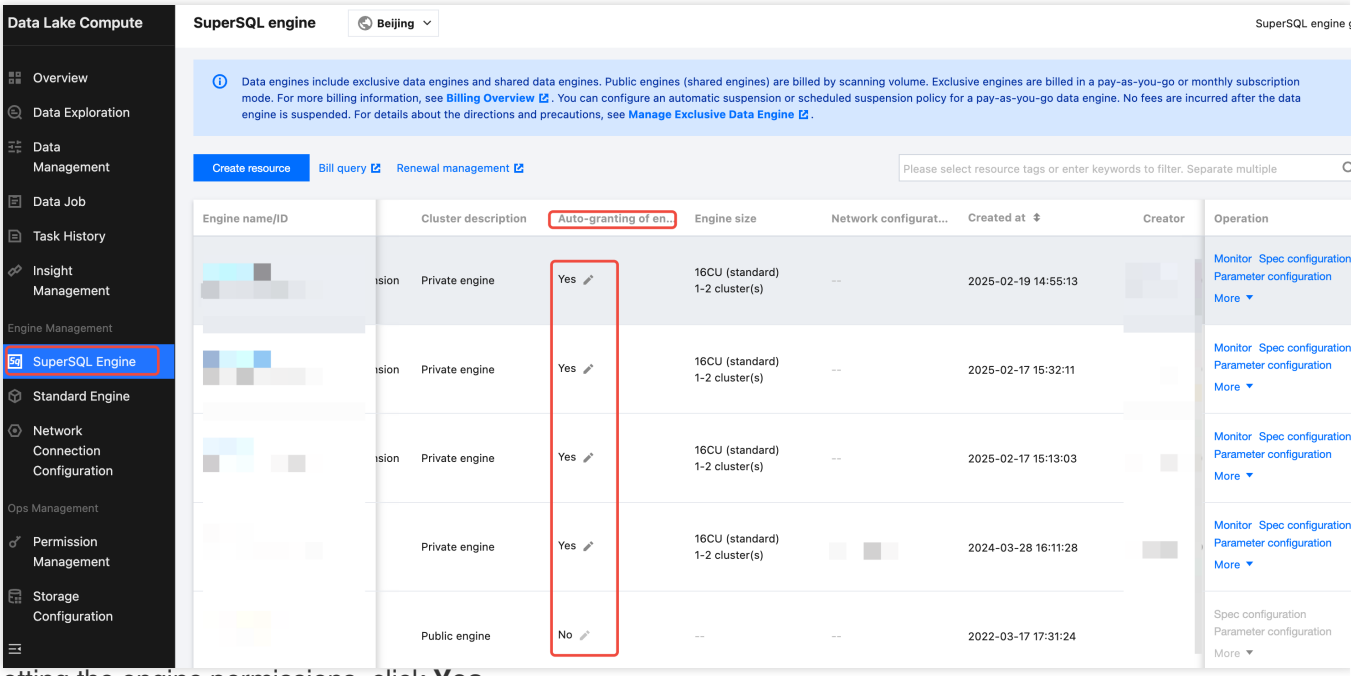
How To Enable and Disable Automatic Authorization Engine Permissions

There are two permission entries for default enabling/disabling engine operation:

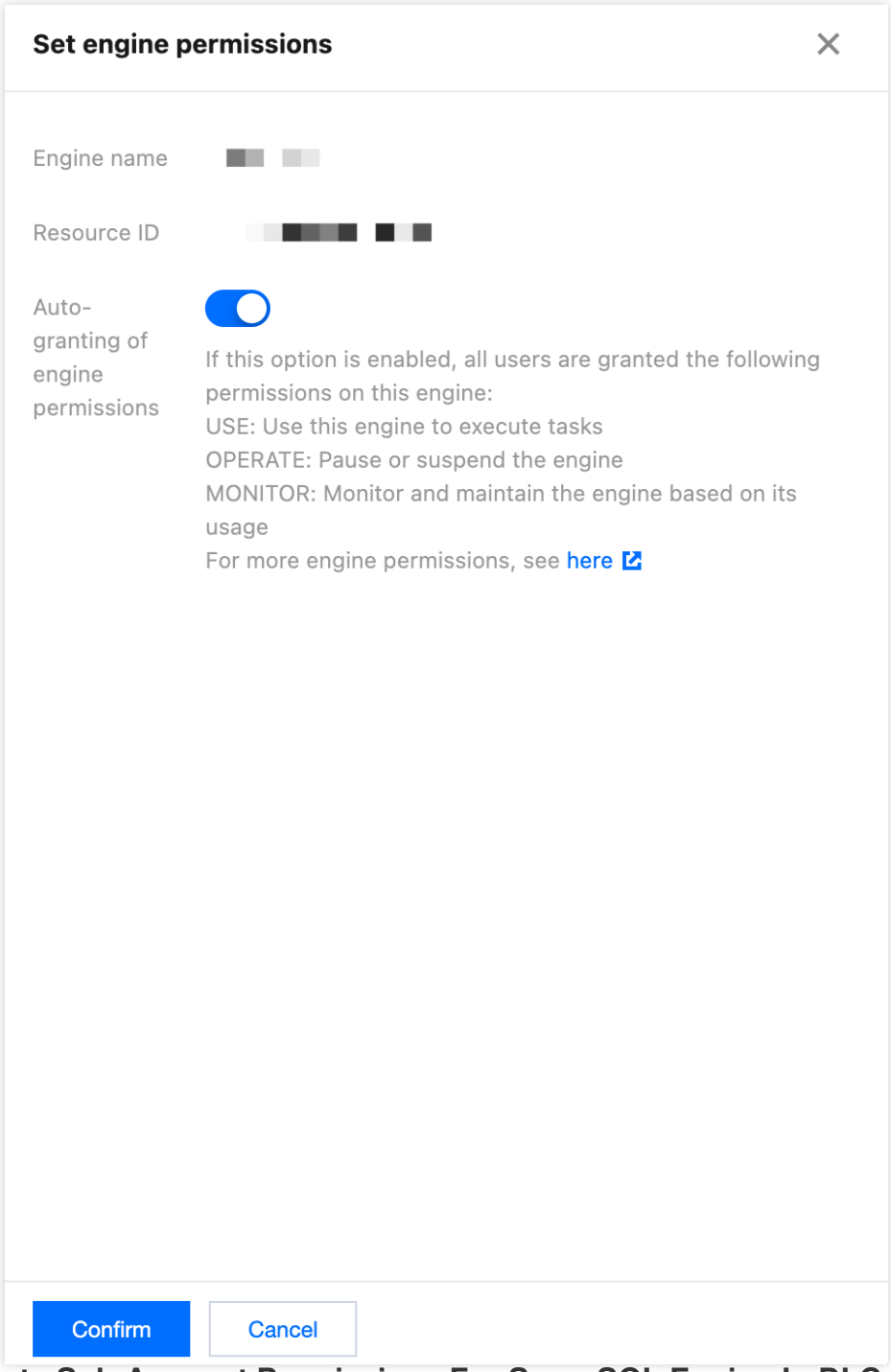
Entry 1:**Engine Purchase Page** >**Advanced Configuration Item.**



Entry 2: Go to the Engine Management page and click to edit authorization engine permissions.



After setting the engine permissions, click **Yes**.



Activate Sub-Account Permissions For SuperSQL Engine In DLC

After creating a user or workgroup, click the authorization operation in the list to add permissions to the workgroup. DLC Data Engines are divided into two categories: **SuperSQL Engine** and **Standard Engine**. For detailed differences and application scenarios, please refer to Data Engines. The earlier-launched **SuperSQL Engine** permissions are managed through the DLC console, where you can quickly manage SuperSQL Engine permissions in DLC Console > Permission Management. On the other hand, **Standard Engine's** permission management is uniformly controlled by Tencent Cloud CAM. For information on Standard Engine's permission management, please refer to DLC Permission Overview.

For the SuperSQL Engine, based on the usage scenario of the user or workgroup, you can check the engine's permission policy in **DLC Console > Permission Management > Engine Permissions**.

Note:

Usage: Use this engine for task execution.

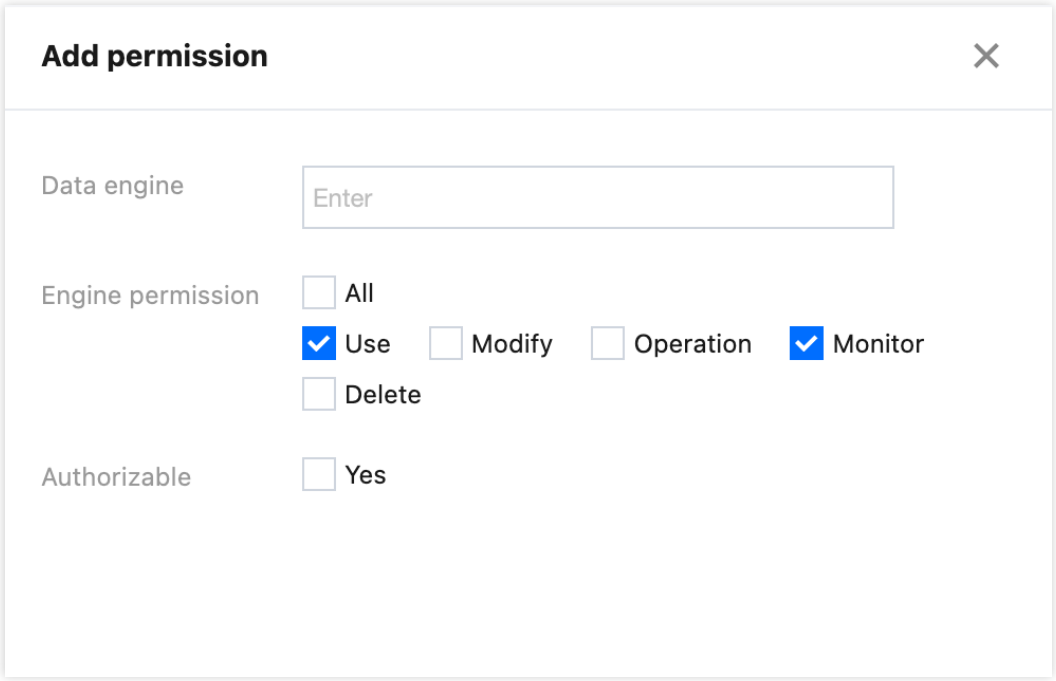
Modification: Modify the engine's configuration parameters, such as specification adjustment of the engine.

Operation: Suspend or hang up the engine.

Monitoring: Monitor the usage and Ops of the engine.

Deletion: Delete the engine.

Authorizable: After checking, all members under this Sub-user or workgroup have the authorization permission for the engine.



Data Permission Management

Note:

Operation permission: Configured with the root account or a DLC administrator account.

The root account or DLC administrator account can be used to configure permissions based on the usage scenario of users or work groups. On the Permission Management page, select the data permission policy.

DLC data permissions include:

Data directory permissions

Permission for creating databases under the DataLakeCatalog directory and the permission for creating databases in other data directories.

Permission scope:

1. Whether to allow users or work groups to create databases under DataLakeCatalog
2. Whether to allow users or work groups to create databases in other data directories

Database and table permissions

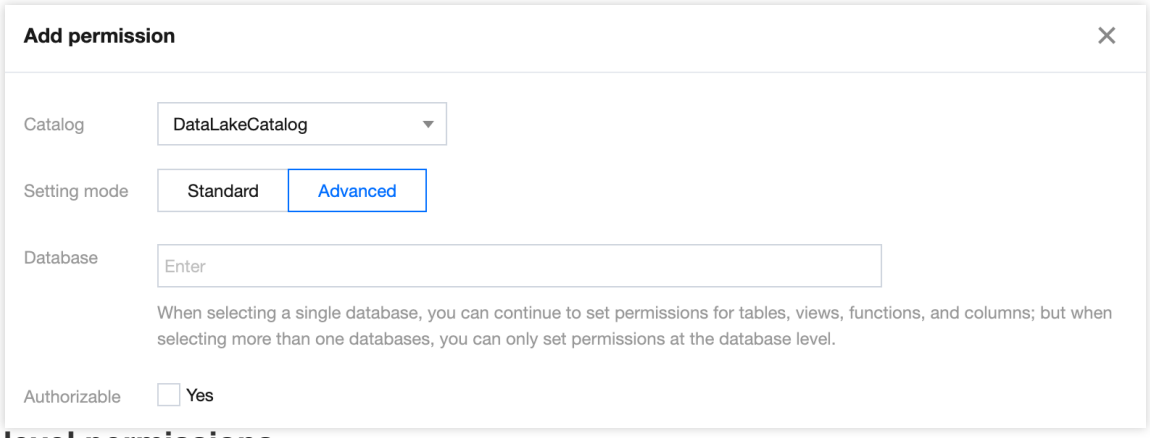
Permissions for setting databases, data tables, views, and functions.

Permission scope:

Permission Type	Databases	Data Tables	Views and Functions
Query and analysis permissions	Permissions to query all tables, views, and functions. create data tables in the database.	Query	Query
Data editing permissions	Permissions to modify and delete databases and create tables. All permissions of all tables, views, and functions.	Data query, insertion, update, and deletion. Table modification and deletion.	Query, creation, modification, and deletion
Owner permissions (permission is further granted based on the data editing permission)	Permissions to modify and delete databases and create tables. All permissions of all tables, views, and functions.	Data query, insertion, update, and deletion. Table modification and deletion.	Query, creation, modification, and deletion

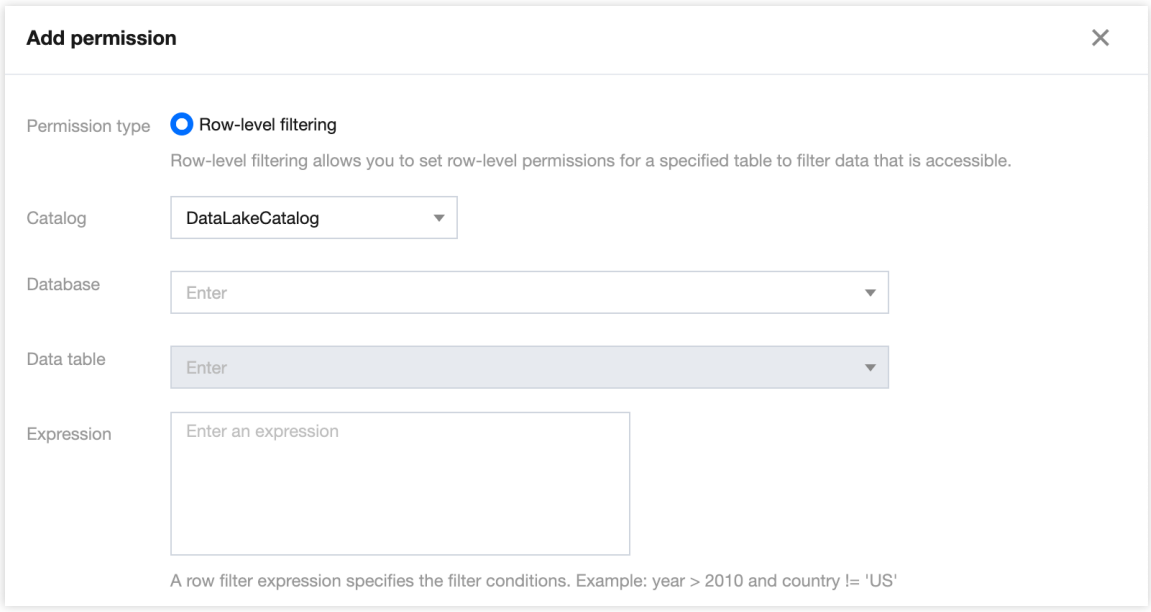
Advanced permission settings for databases and tables

If you select a single database, you can further set the query, insertion, update, and deletion permissions of tables, views, and functions under the database. If you select multiple databases, you can only set database permissions. In advanced mode, you can set column-leve permissions. By selecting a single data table, you can add query permission for columns. You can also select one or more columns or all columns to grant permissions.



Row-level permissions

Based on database and table permissions, add row-level filter expressions to restrict the access scope.



DLC User Management

Note:

Operation permission: Configured with the root account or a DLC administrator account.

Modifying User Permissions

Select a user in the user list and click Edit to manage the user permissions.

Viewing User Permissions

Click a user ID in the user list to enter the user details page.

Deleting User Permissions

Remove specified permissions: In the user list on the Permission Management page, click Edit to enter the details page for modification.
Delete users: In the user list on the Permission Management page, click Delete.

Work Group

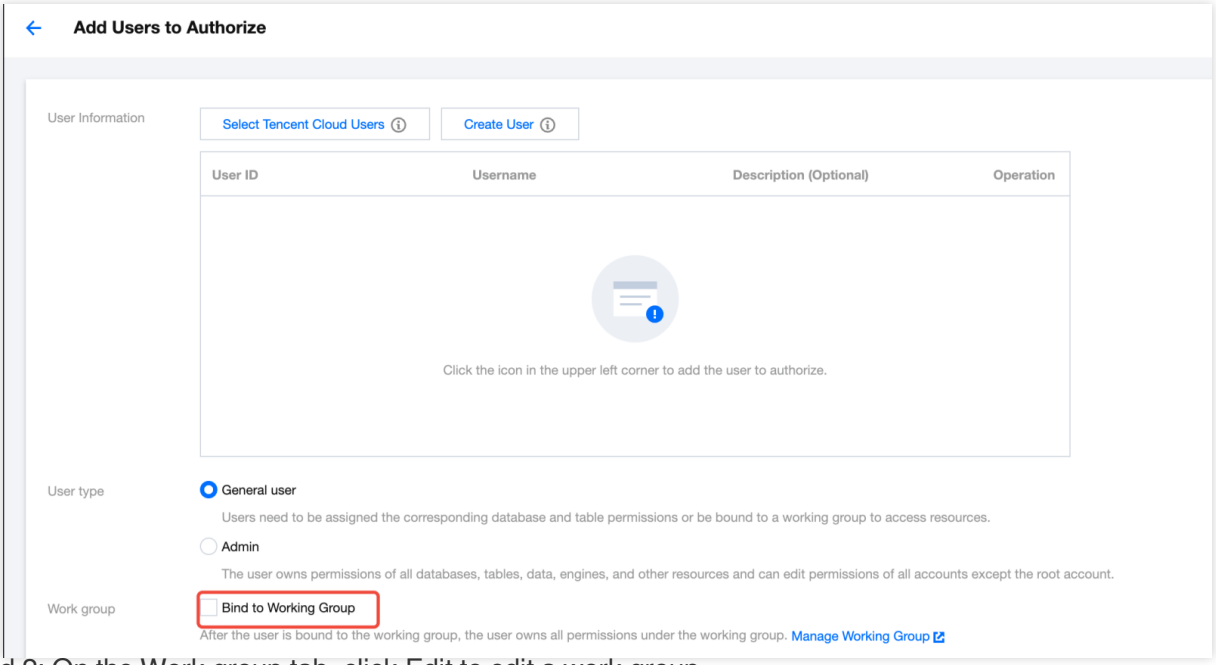
Note:
Operation permission: Configured with the root account or a DLC administrator account.
To achieve better unified management and reduce administrators' management costs, you can create a work group to add users in batch and authorize them in a unified manner.

Creating a Work Group

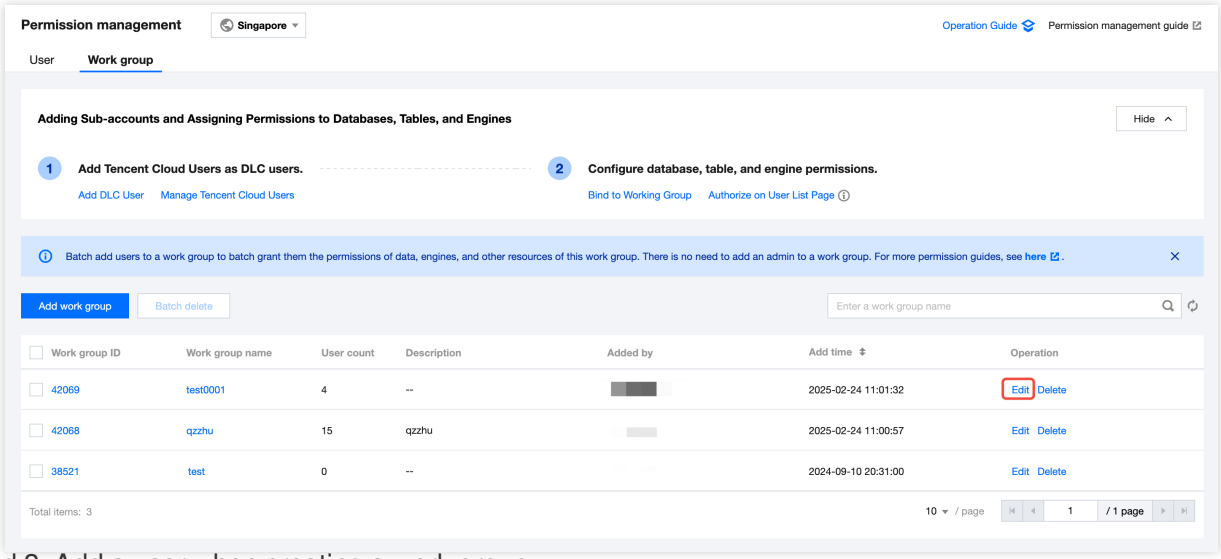
Click **Work group** on the Permission Management page and click Add work group.

Adding Users to a Work Group

Method 1: When adding a user, select Bind to Working Group.



Method 2: On the Work group tab, click Edit to edit a work group.



Method 3: Add a user when creating a work group.

Adding Permissions to a Work Group

On the Work group tab, click Edit and add permissions.

Editing Work Group Members or Permissions

On the Work group tab, click Edit and manage the work group members or permissions.

Deleting a Work Group

On the Work group tab, click Delete for the work group.

FAQs

Why a failure occurs when I add the account as a DLC account?

If an error occurs when you add the account as a DLC account, first check whether any of the following issues occur. If not, please contact aftersales personnel for help.

1. Repeated addition: The account has already been added as a DLC account.
2. Account input error: An incorrect account is entered.

Why it displays “unavailable feature” or “no permission” when I enter the DLC console?

Your account may not have been added as a DLC account, or the administrator has not configured permissions for your account. You can contact your administrator to add your account as a DLC account and configure the relevant permissions for your account.

Which operations must be performed with the root account during the use of DLC?

The root account is responsible for activating the DLC service, creating Tencent Cloud sub-accounts, authorizing the policy QcloudDLCFullAccess for sub-accounts, designating the first DLC administrator, and granting financial permissions.

How are the permissions determined if the permissions of a user are inconsistent with those of the work group to which the user belongs?

The permissions are the union of user and work group permissions.

DLC Data Import Guide

Last updated：2024-07-31 17:23:10

External Table Data Import via COS

DLC supports querying and analyzing data directly on COS without migrating the data. Therefore, you only need to import the data into COS to start using DLC for seamless data analysis, achieving complete decoupling of data storage and computation. Currently, it supports uploading in multiple formats such as orc, parquet, avro, json, csv, and text files.
Currently, COS offers a variety of data import methods. You can choose from the following methods based on your situation.
log in to COS and proceed with file upload directly. For related operating steps, see Uploading an Object.
Import data using various upload tools provided by COS. For a list of supported tools, see Tool Overview.
Import data using SDKs or APIs provided by the COS service. For service-related instructions, see Upload Interface Documentation.
If you need to analyze logs from CLS, you can directly deliver logs to COS by partition and then analyze and query directly through DLC. For related operations, see Using DLC (Hive) to Analyze CLS Logs.
If you need to import data from other cloud services (such as database CDB, etc.) into COS, you can use DataInLong to perform the import. When creating a data synchronization link, select the cloud service to export from as the data source and choose COS as the destination to complete the data import.
If you encounter any issues during data import, you can consult us for a solution by Submitting a Ticket.
After importing data into COS, you can perform SQL queries through the DLC console, API, or SDKs, enabling table creation, analysis, and export of results. For detailed operations, see Quick Start with Data Analytics in Data Lake Compute.

Data import into native tables

To provide better data query performance, DLC also supports importing data into native tables for query analysis. DLC native tables are arranged in the Iceberg table format, optimizing data during the import process. If you have the following use cases, it is recommended to use native tables for data query analysis.
In data warehouse analysis scenarios, aiming to leverage the Iceberg index for better analytical performance.
If there's a need to update data, the DLC service supports performing UPSERT operations through SQL or data jobs.
Data is written or updated in real-time through DataInLong, Flink, SCS, Spark Streaming, with concurrent reads and writes, requiring transactional guarantees for data processing business.

Wishing to utilize Iceberg table features, such as time travel, multi-version snapshots, hidden partitions, partition evolution, and other advanced data lake features.
If you need to import data into a native table, you can choose one of the following methods based on your situation.
Directly import through the DLC console.
Caution
When importing data through the console, there are certain restrictions, mainly for rapid testing and it's not recommended for production use.
If your original data is in services like MySQL or Kafka and you need to write or update MySQL binlog and message middleware data to DLC in near real-time, this can be achieved through DataInLong DataInlong's real-time import capability. Or through SCS, Flink writing. For operational guidance, you can contact us through a Work Order.
If the original data is in data services such as MySQL, Kafka, MongoDB, etc., offline synchronization tasks by DataInLong DataInLong can be used to transfer data to native tables. During the data warehouse modeling process, external tables are used as the source layer of original data. In the process of transferring data to native tables, business-specific data distributions can be reorganized through building sparse indexes, etc., to achieve excellent query analysis performance of native tables. If guidance is needed, you can Contact Us.
Use SQL statements SELECT INSERT to query the data from the external table and then write it into the native table. For example, after creating a native table in DLC with the same table structure as the external table, the transfer can be completed by executing SQL syntax with the SparkSQL engine. Syntax example is as follows:

```
--- External table name: outtertable, Native table name: innertable
insert into innertable select * from outtertable
```

If you encounter any issues during data import, you can consult us for solutions by submitting a work order.

Multiple data sources federated query analysis

If you do not wish to export data to the native tables of COS or DLC, DLC also offers the capability of data federation query analysis. It supports rapid association and analysis of data from multiple data sources through SQL without relocating data. Currently, it supports a variety of data sources including MySQL, SQLServer, clickhouse, PostgreSQL, EMR on HDFS, and EMR on COS.
When using federated analysis, it is necessary for the data source and data engine to be on the same network, ensuring network connectivity. Management can refer to Engine Network Configuration.
When querying EMR data through DLC federated analysis, the query performance will be on par with or even exceed that of EMR, making it suitable for production environments. It allows for the full utilization of DLC's fully-managed elastic capabilities to reduce costs and increase efficiency without relocating EMR services.
Federated analysis enables quick unification and analysis of data from multiple data sources, providing a convenient method for data insights and rapid analysis. With the support of DLC's fully-managed elastic capabilities, it effectively

reduces the cost of use. It also supports the use of INSERT INTO/INSERT OVERWRITE syntax to write federated data into DLC native tables, completing data import.
When analyzing data from other data sources through federated analysis, since the computation process involves synchronizing data to the DLC for analysis, there is some performance loss compared to directly querying the original data sources. If high query performance is required, data can be imported into native tables for analysis. The operation can be seen in Data import into native tables.

Quick Start with Data Analytics in Data Lake Compute

Last updated：2024-07-17 15:19:00

Data Lake Compute allows you to quickly query and analyze COS data. Currently, CSV, ORC, Parquet, JSON, Avro, and text files are supported.
With Data Lake Compute, you can complete data analysis queries on COS in just a minute. It currently supports multiple formats including CSV, ORC, PARQUET, JSON, ARVO, and text files.

Preliminary Preparations

Before initiating a query, you need to activate the internal permissions of Data Lake Compute and configure the path for query results.

Step 1: Establish the necessary internal permissions for Data Lake Compute.

Note

If the user already has the necessary permissions, or if they are the root account administrator, this step can be disregarded.
If you are logging in as a sub-account for the first time, in addition to the necessary CAM authorization, you also need to request any Data Lake Compute admin or root account admin to grant you the necessary Data Lake Compute permissions from the **Permission Management** menu on the left side of the Data Lake Compute console (for a detailed explanation of permissions, please refer to DLC Permission Overview).
1. Table Permissions: Grant read and write operation permissions to the corresponding catalog, database, table, and view.
2. Engine Permissions: These can grant usage, monitoring, and modification rights to the computation engine.

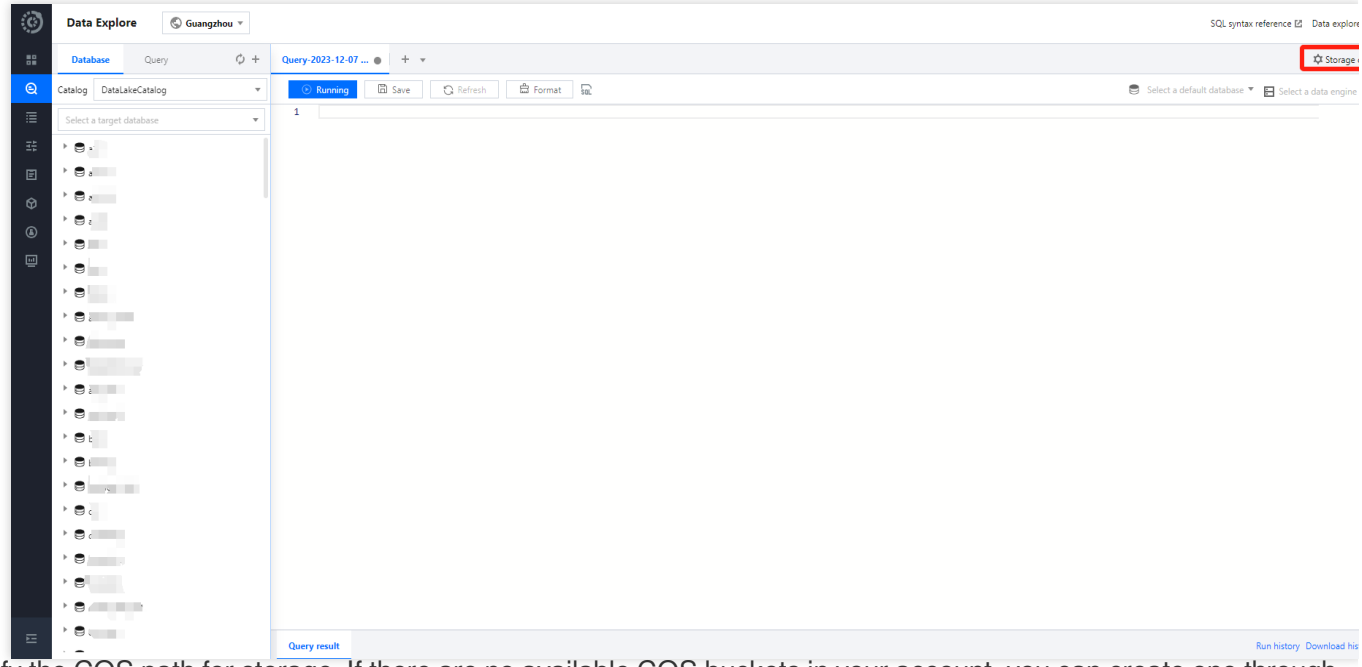
Note

The system will automatically provide each user with a shared public-engine based on the Presto kernel, allowing you to quickly try it out without the need to purchase a private cluster first.
For detailed steps on granting permissions, please refer to Sub-account Permission Management.

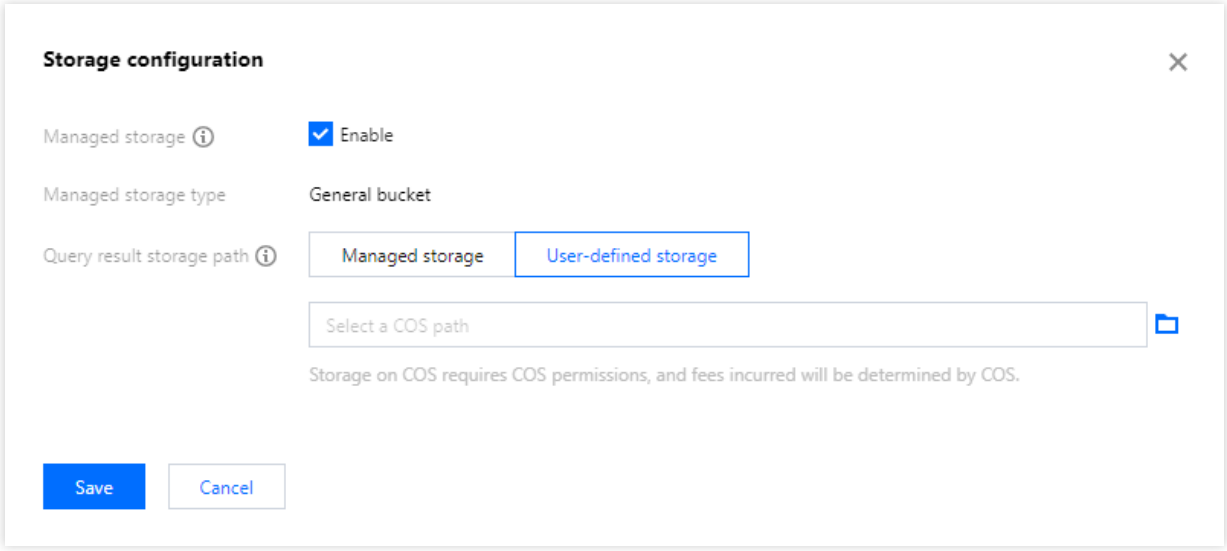
Step 2: Configure the path for query results.

Upon initial use of Data Lake Compute, you must first configure the path for query results. Once configured, the query results will be saved to this COS path.
1. Log in to the Data Lake Compute DLC console and select the **service region**.
2. Navigate to **Data Exploration** via the left sidebar menu.

3. Under the **Database and Tables** page, click on **Storage Configuration** to set the path for query results.



Specify the COS path for storage. If there are no available COS buckets in your account, you can create one through the Object Storage Console.

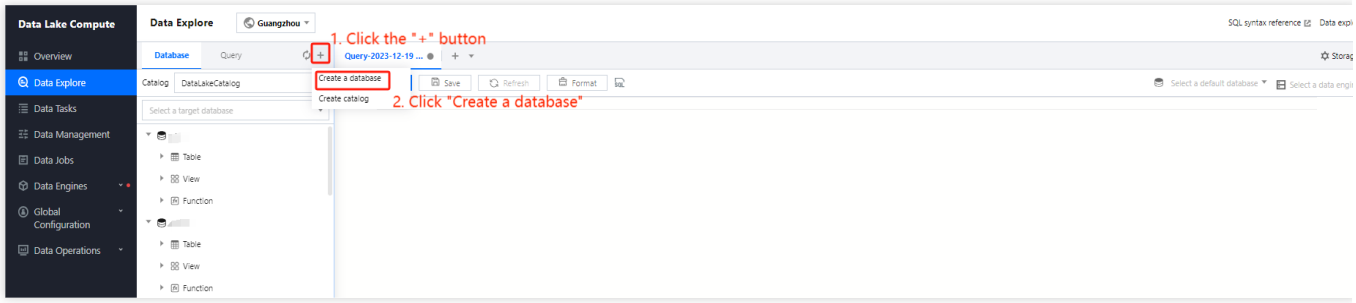


Analysis Steps

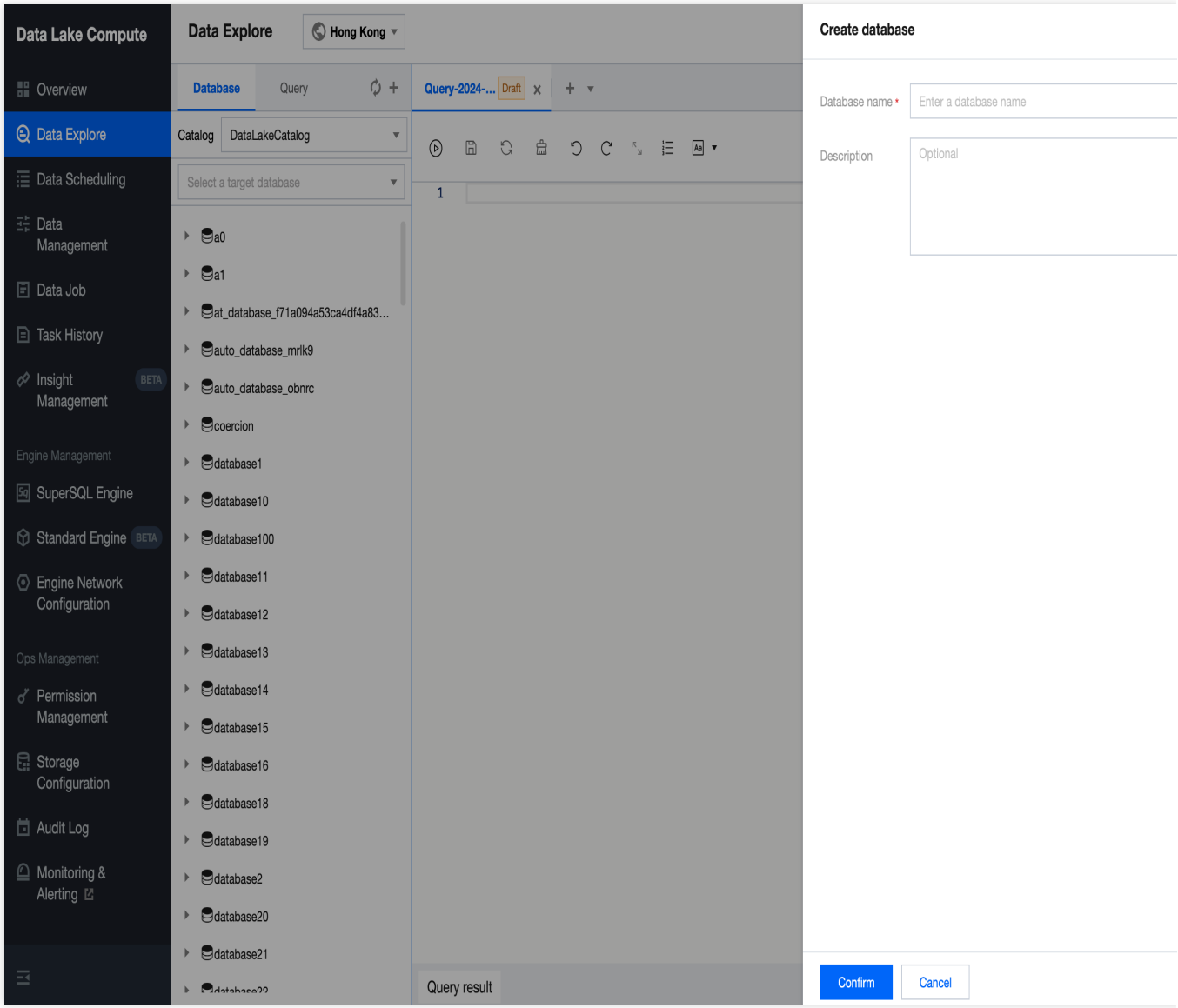
Step 1. Create a database

If you are familiar with SQL statements, write the `CREATE DATABASE` statement in the query and skip the creation wizard.

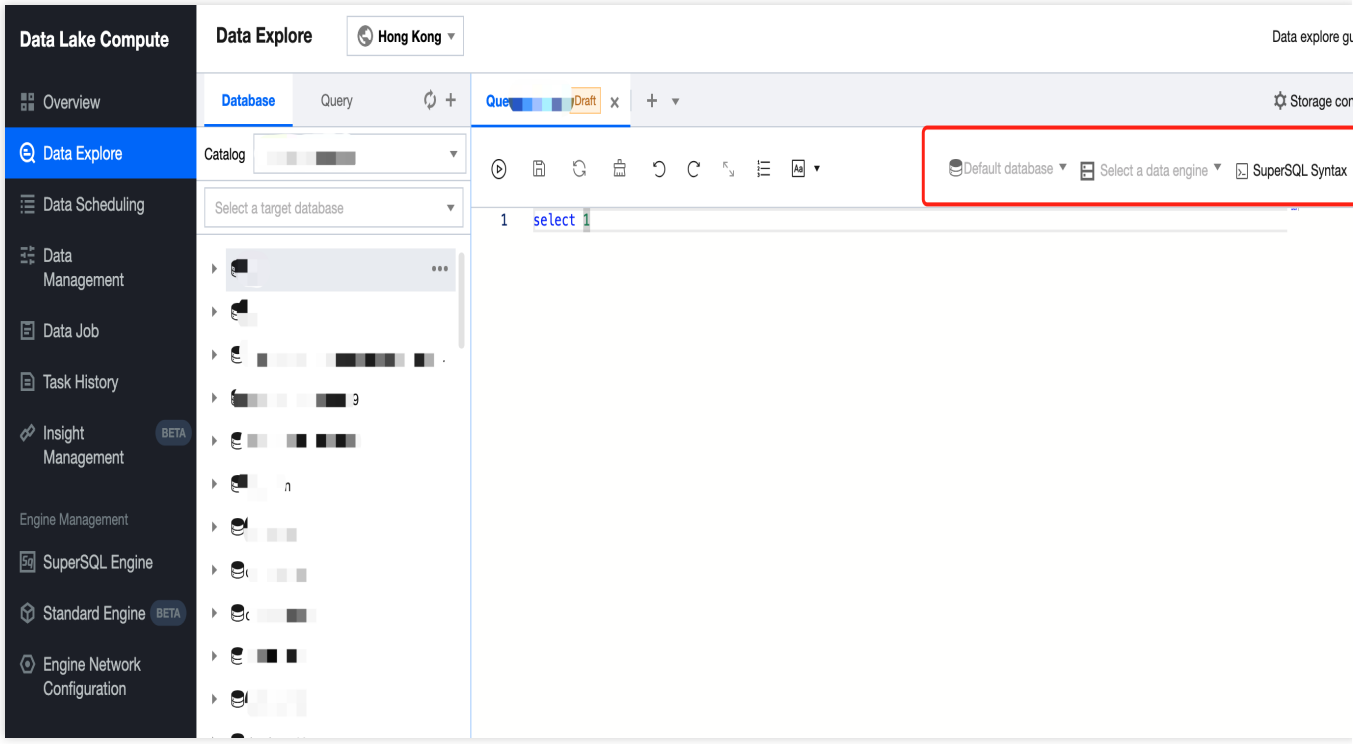
1. Log in to the Data Lake Compute console and select the service region.
2. Select **Data Explore** on the left sidebar.
3. Select **Database & table**, click "+", and select **Create a database** as shown below:



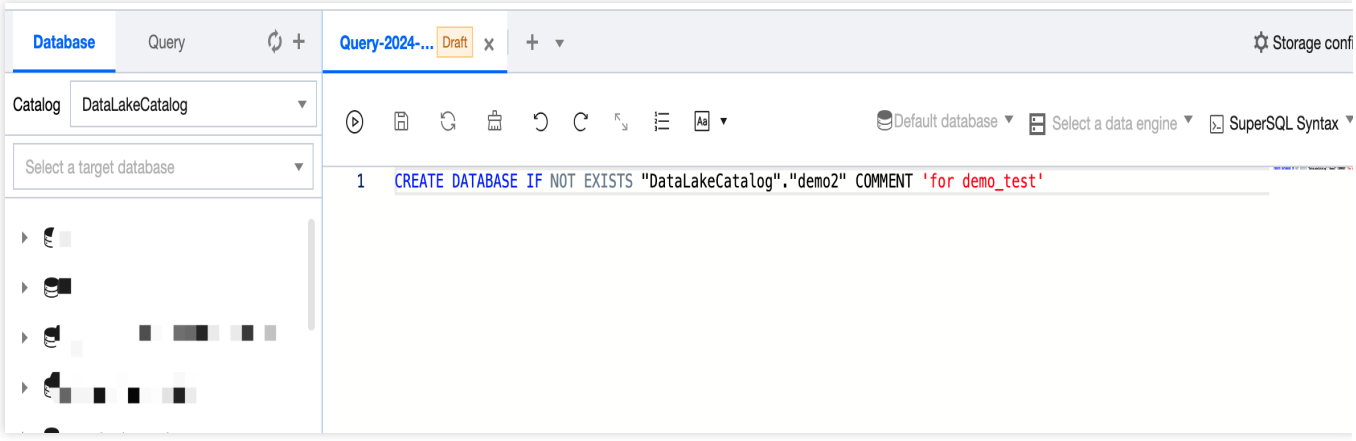
Enter the database name and description.



4. After selecting an execution engine in the top-right corner, run the `CREATE DATABASE` statement.



As shown in the picture below:



For details, see Table Management.

Step 2. Create an external table

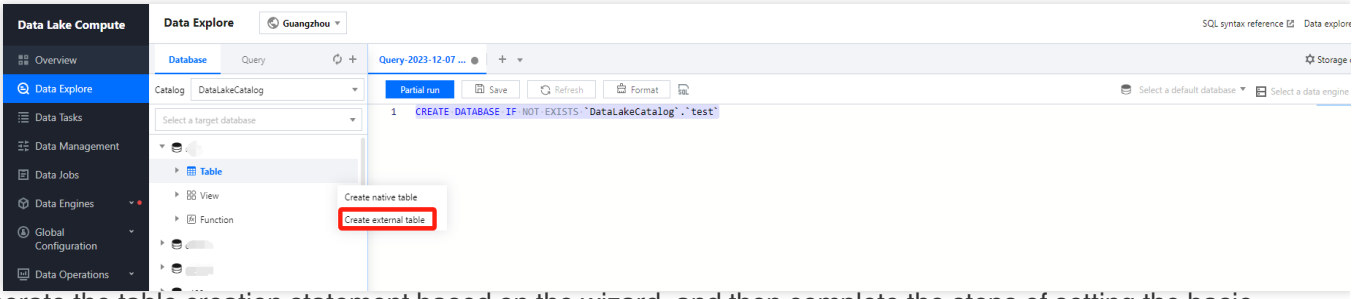
If you are familiar with SQL statements, write the `CREATE TABLE` statement in the query and skip the creation wizard.

1. Log in to the Data Lake Compute console and select the service region.
2. Select **Data Explore** on the left sidebar.

3. Select **Database & table**, select the created table, and right-click to select **Create external table**.

Note:

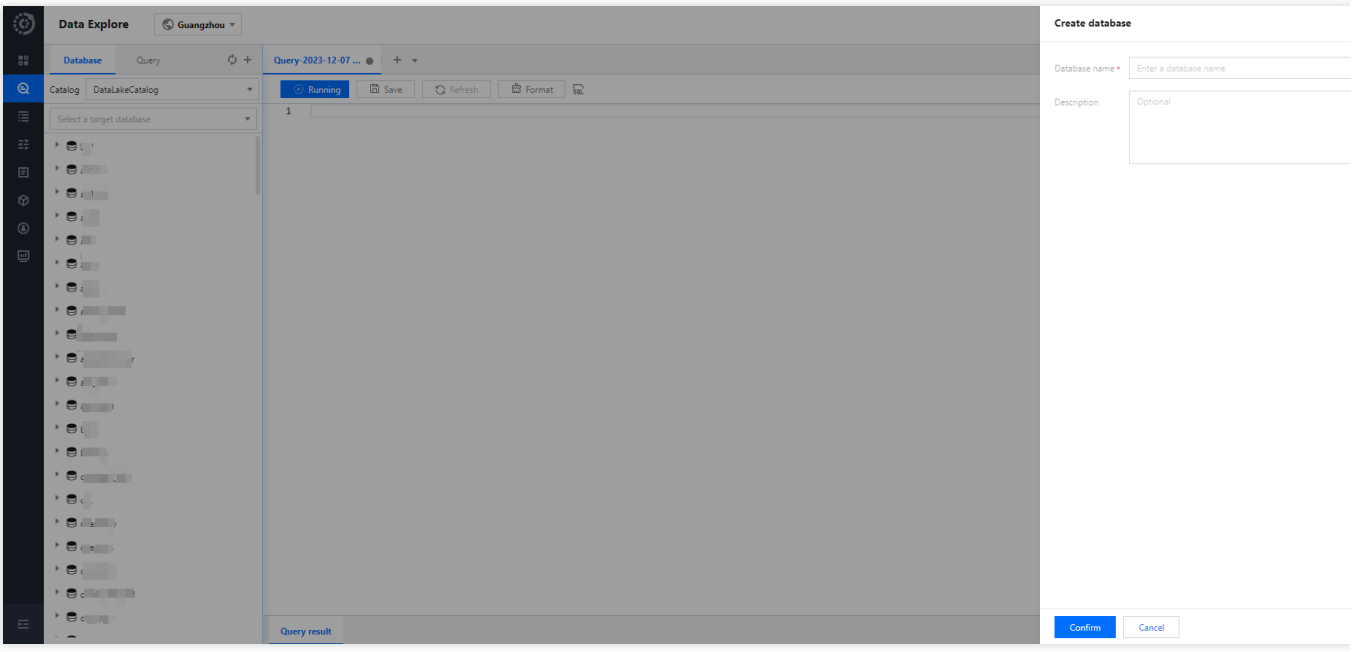
An external table generally refers to a data file stored in a COS bucket under your account. It can be directly created in Data Lake Compute for analysis with no need to load additional data. It is external, so only its metadata will be deleted when you run `DROP TABLE`, while your original data will remain.



4. Generate the table creation statement based on the wizard, and then complete the steps of setting the basic information, selecting the data format, editing the column, and editing the partition.

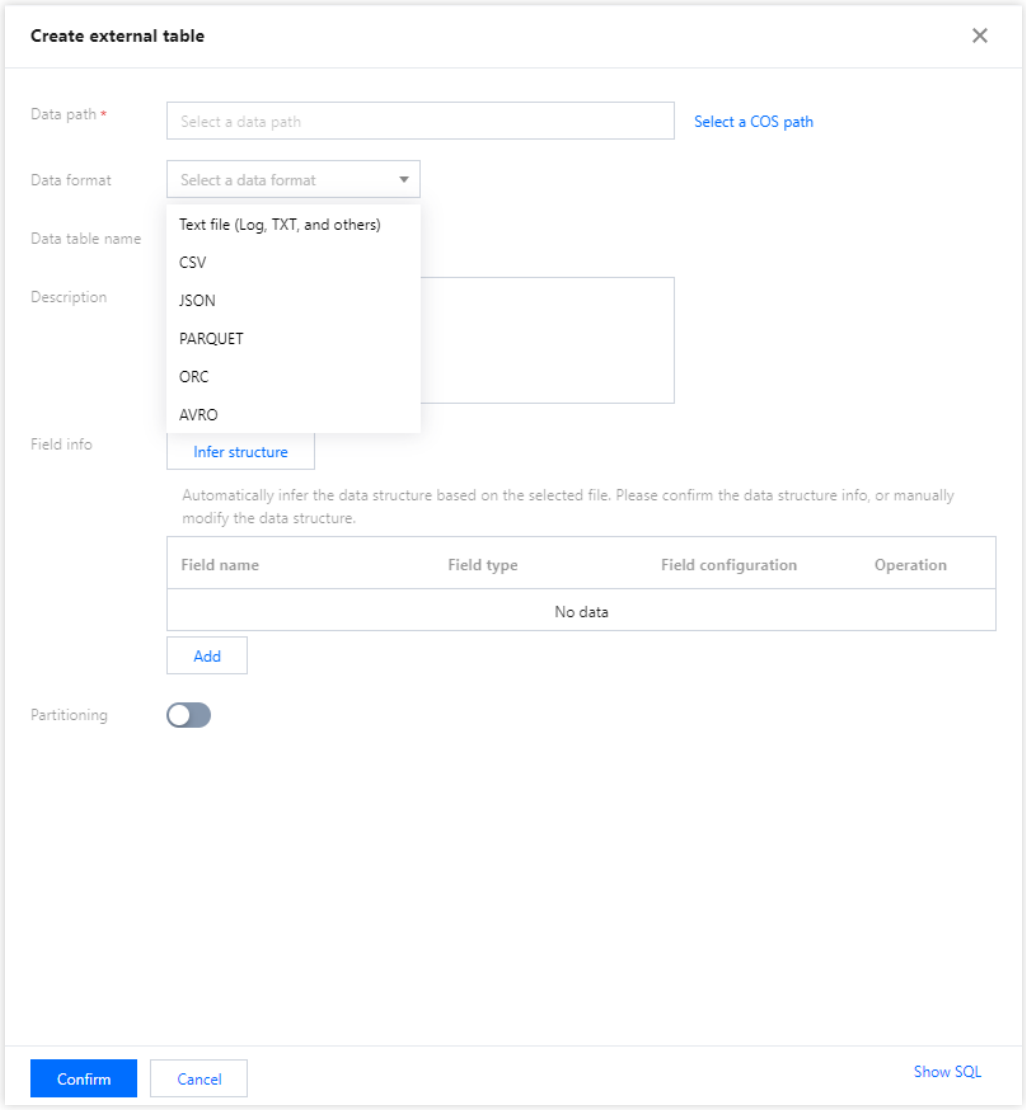
Step 1. Select the COS path of the data file (which must be a directory in a COS bucket but not a bucket itself). There is also a quick method to upload a file to COS. The operations require relevant COS permissions.

Step 2. Select the data file format. In the **Advanced options**, you can select automatic inference, and then the backend will parse the file format and automatically generate the table column information for fast column inference.

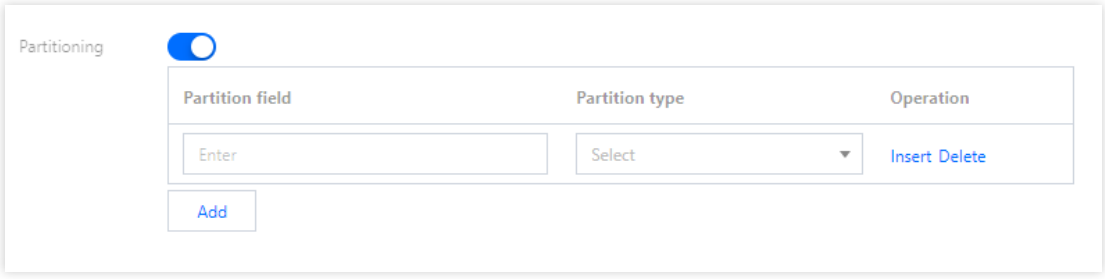


Note:

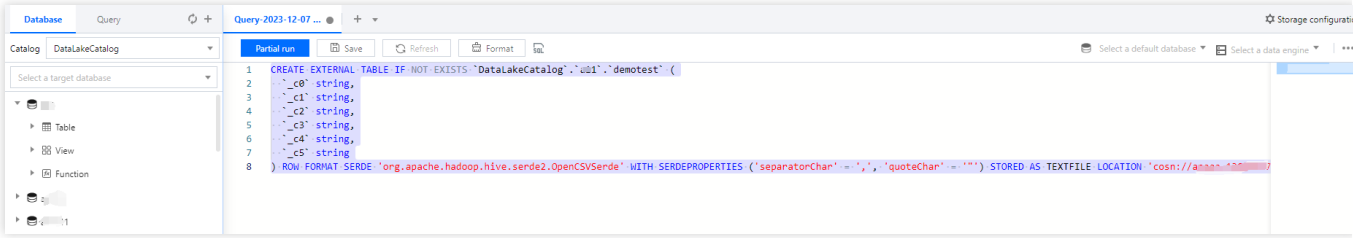
Structure inference is an auxiliary tool for table creation and may not be 100% accurate. You need to check and modify the field names and types as needed.



Step 3. Skip this step if there is no partition. Proper partitioning helps improve the analysis performance. For more information on partitioning, see Querying Partition Table.

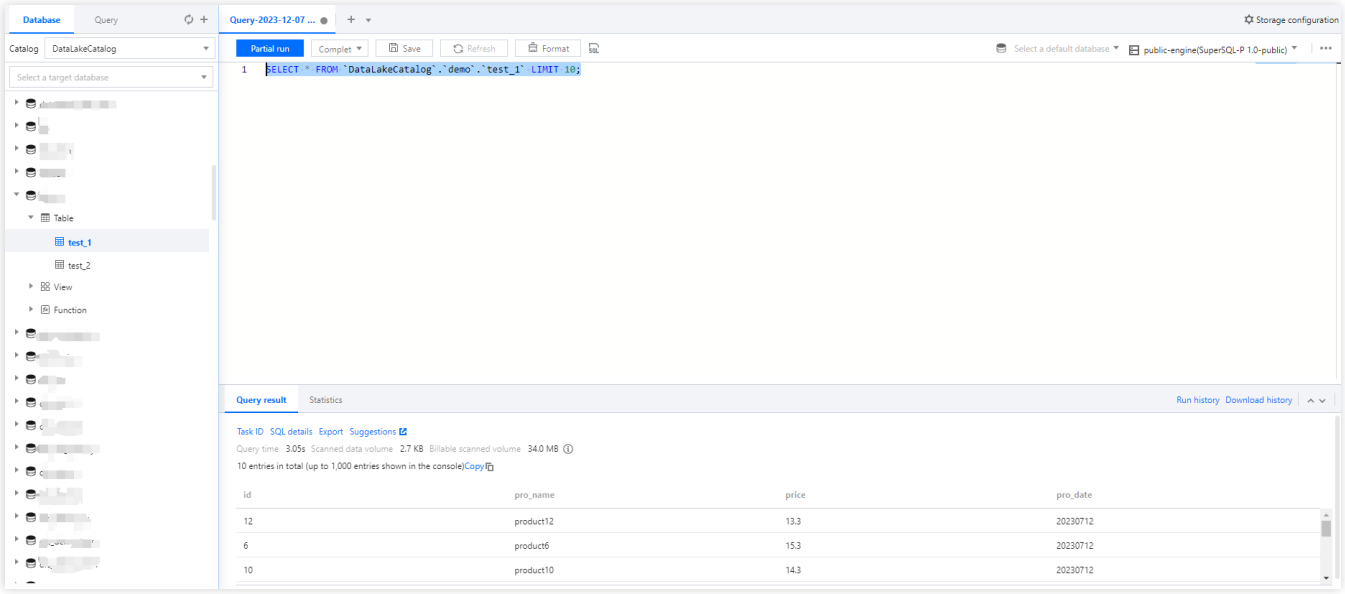


5. Click **Complete** to generate the SQL statement for table creation. Then, select a data engine and run the statement to create a table.



Step 3. Run the SQL analysis

After the data is prepared, write the SQL analysis statement, select an appropriate compute engine, and start data analysis.



Sample

Write a SQL statement with all data query results being `SUCCESS` and run the statement after selecting a compute engine.

```
select * from `DataLakeCatalog`.`demo2`.`demo_audit_table` where _c5 =
'SUCCESS'
```

Quick Start with Permission Management in Data Lake Compute

Last updated：2024-09-18 18:02:02

During the utilization of Data Lake Compute (DLC), if you need to establish varying access permissions for employees within your organization to achieve isolation of authority among them, you can employ the permissions management feature for meticulous management of user and workgroup permissions.

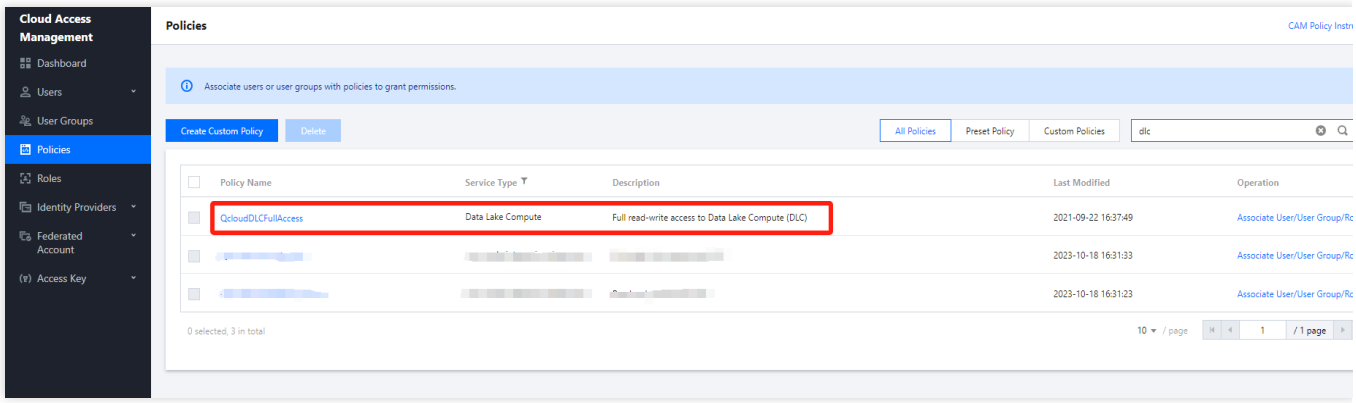
Note：

1. The policy of permissions is highly correlated with the usage of the product. It is recommended that administrators configure the policies for roles such as workgroups and sub-users in advance before officially utilizing the product features.
2. In different regions, administrators are required to reconfigure the member management and permissions management for DLC in that specific region.

CAM Authorization

Data Lake Compute (DLC) possesses a comprehensive data access permission mechanism. If you have sub-account management requirements, please grant the corresponding sub-account with the QcloudDLCFullAccess (Full read-write access to Data Lake Compute (DLC)) policy in the Access Management Console. For specific steps on creating sub-accounts and authorizing policies.

Data Lake Compute (DLC) offers permissions refined to the granularity of row and column levels in data tables, ensuring that you need not worry about overstepping authority with this operation.



Users and Workgroups

DLC manages user permissions through two methods: user authorization and workgroup binding authorization.

User: Refers to users in CAM, including administrators, sub-accounts, and collaborator accounts.

Workgroup: DLC allows a group of users to be bound to a workgroup, granting the group access to data, engines, and other resources. This enables batch management of user permissions, ensuring that all users within the same workgroup have the same level of access.

Note：

When a user's individual permissions differ from the permissions of the workgroup they belong to, the combined permissions will be the union of both sets.

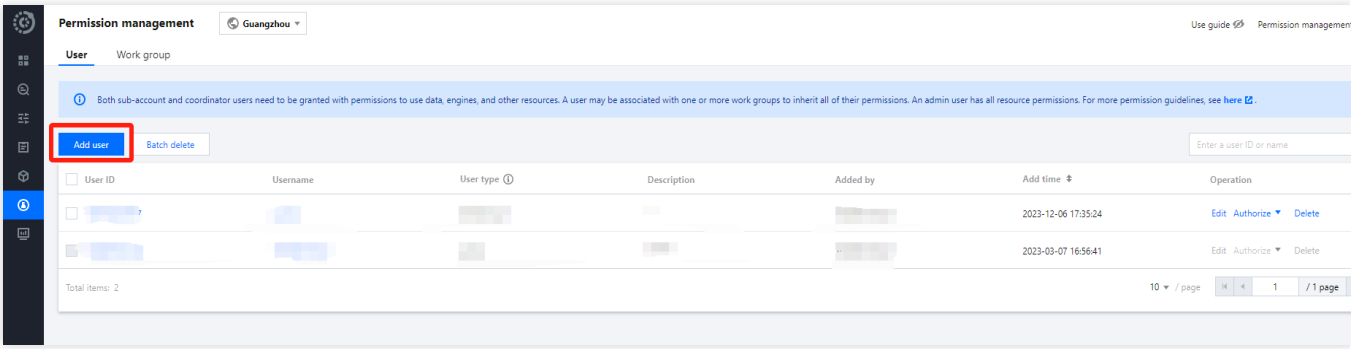
By default, regular users created by an administrator do not have any permissions. To grant permissions, users should be added to a workgroup, and appropriate permission policies should be assigned to the workgroup, allowing the users within it to acquire the necessary permissions.

Adding a User

Data Lake Compute utilizes the Tencent Cloud account ID as the default user ID. It distinguishes between two user types: administrators and ordinary users. Administrators inherently possess all resource permissions, while ordinary users must be granted specific permissions or be associated with a work group to acquire permissions.

1. Incorporate a user and associate them with a work group.

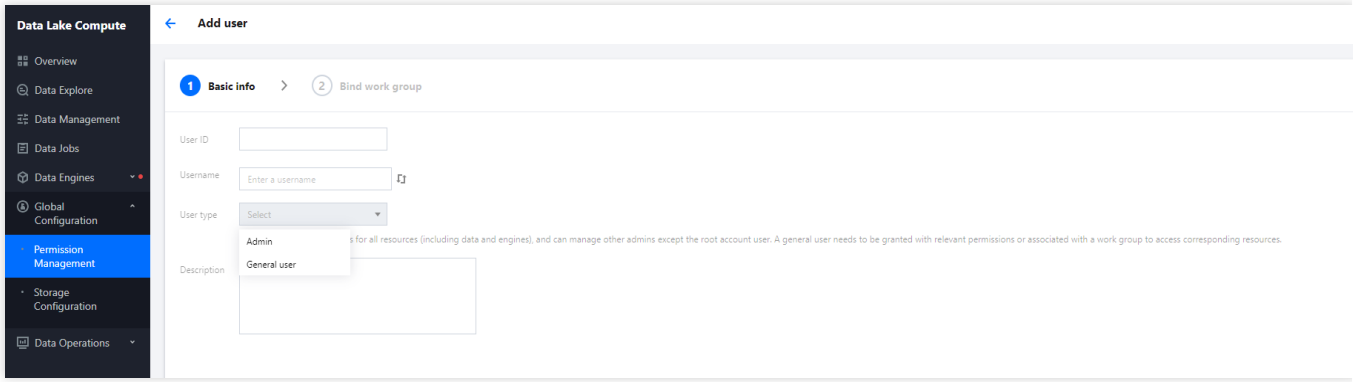
Log into the DLC console, select Permission Management, and click on Users > Add User to incorporate a new user.



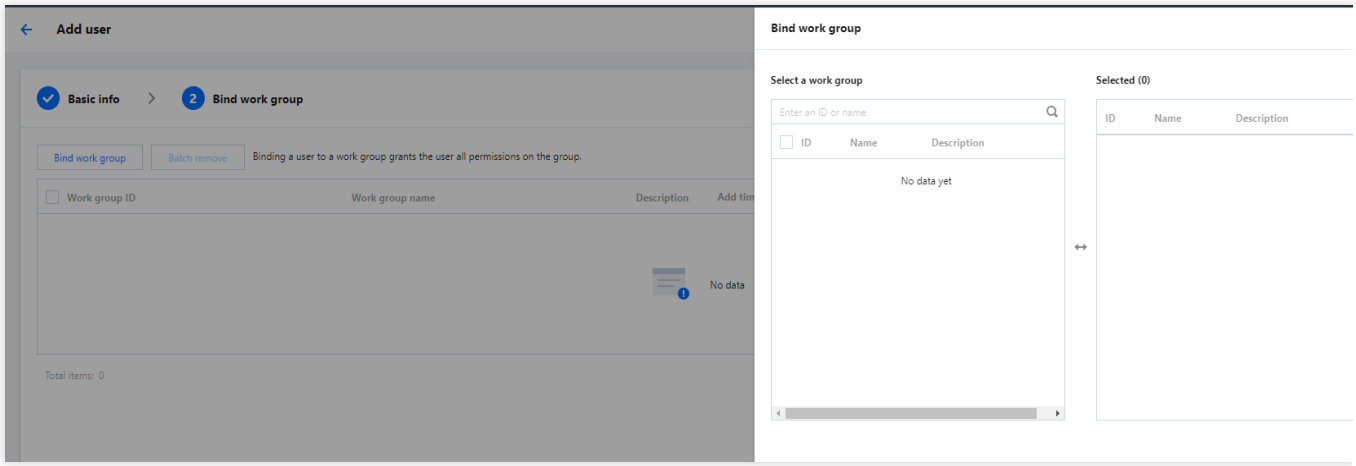
2. Enter the basic information: Provide the user ID, user name, and description, and select the user type.

Note：

When selecting the user type as "Ordinary User", permissions can be obtained through individual authorization or by acquiring all permissions of a specified work group. When selecting "Administrator" as the user type, there is no need to associate with a work group to gain all permissions.

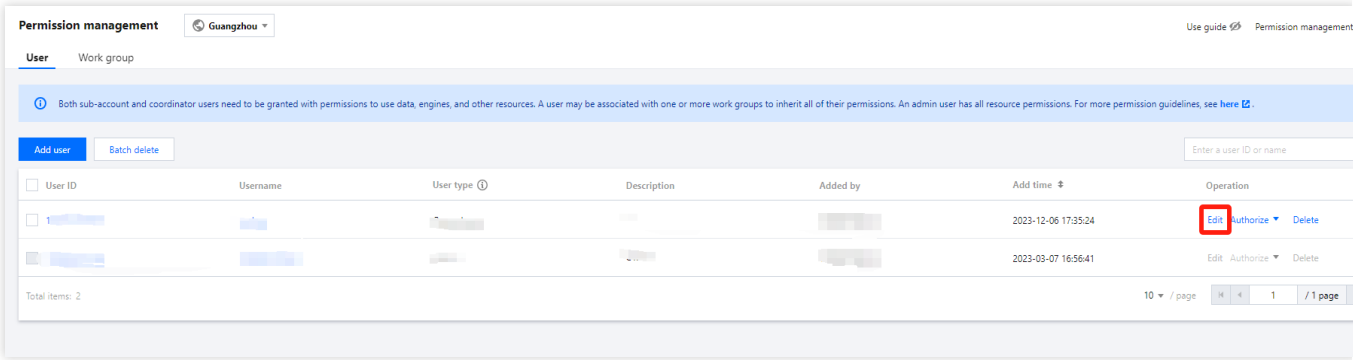


3. Associate with a work group: Select a work group for association (optional).



User authorization

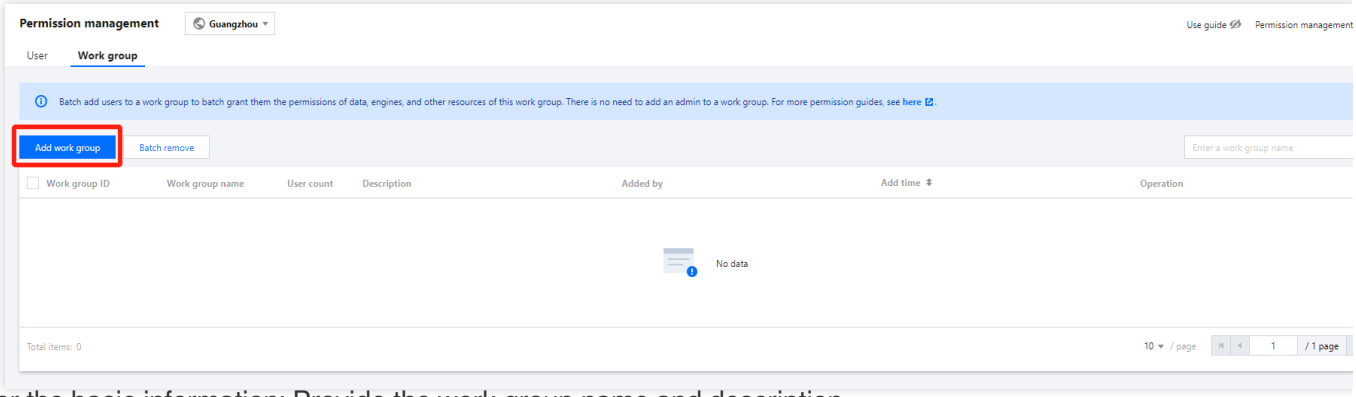
In the user list, authorize each user individually. The authorization includes "Data Permissions" and "Engine Permissions", and the permission policy is consistent with the work group's permission policy.



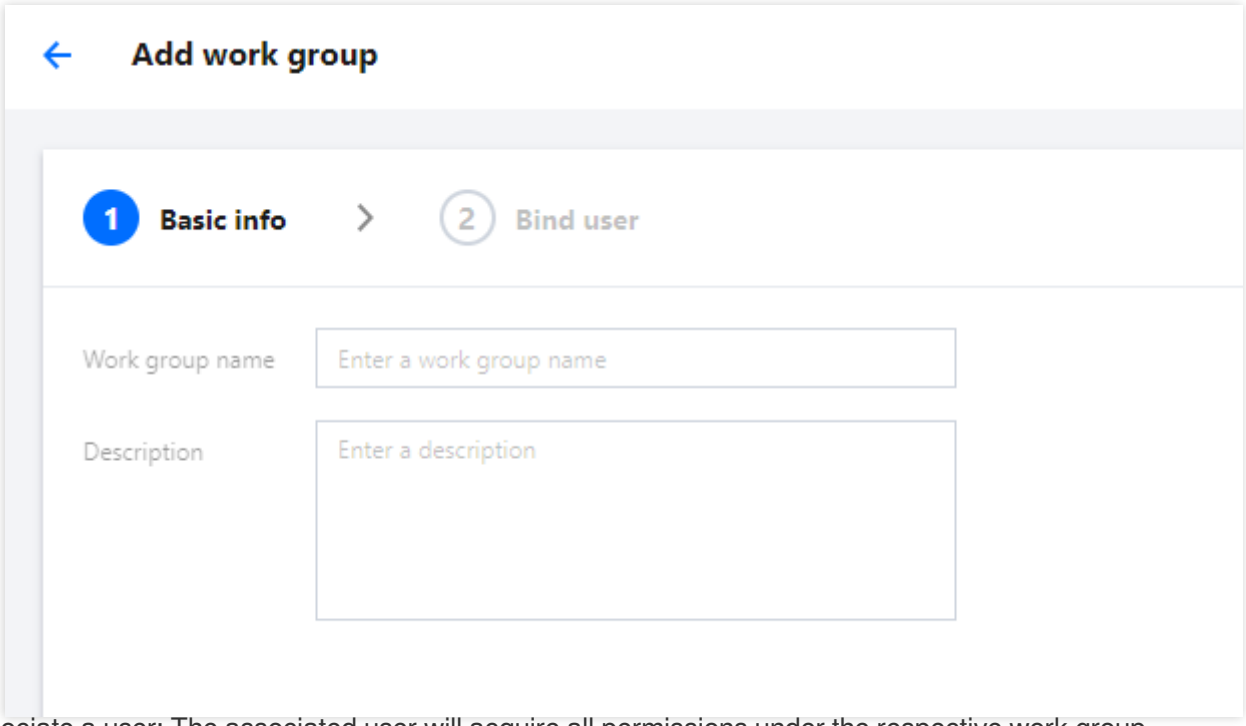
Add Work Group

1. In the Data Lake Compute DLC, select Permission Management from the left sidebar, and click on Work Group > Add Work Group to create a work group for the user. When creating a work group, you can choose to bind it to a user

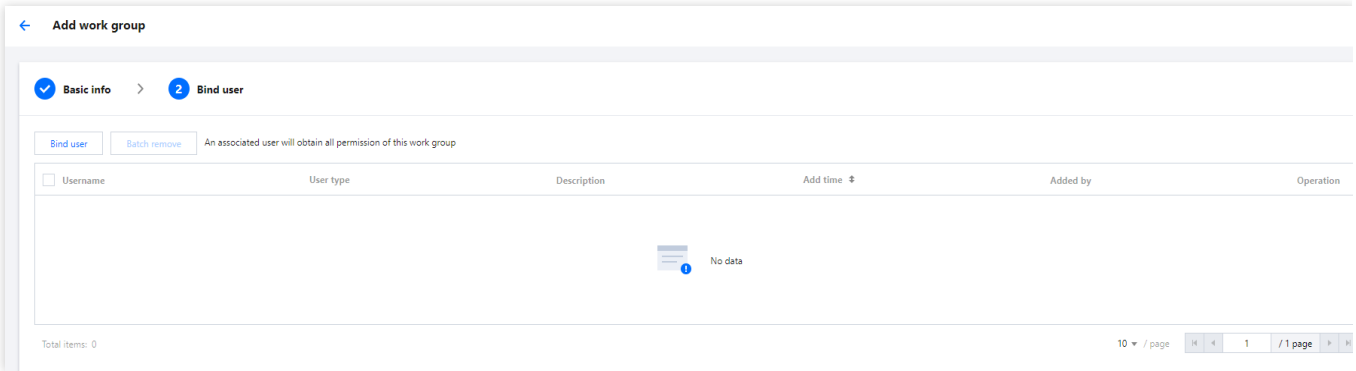
or create an empty work group. For detailed operations, refer to Users and User Groups.



2. Enter the basic information: Provide the work group name and description.

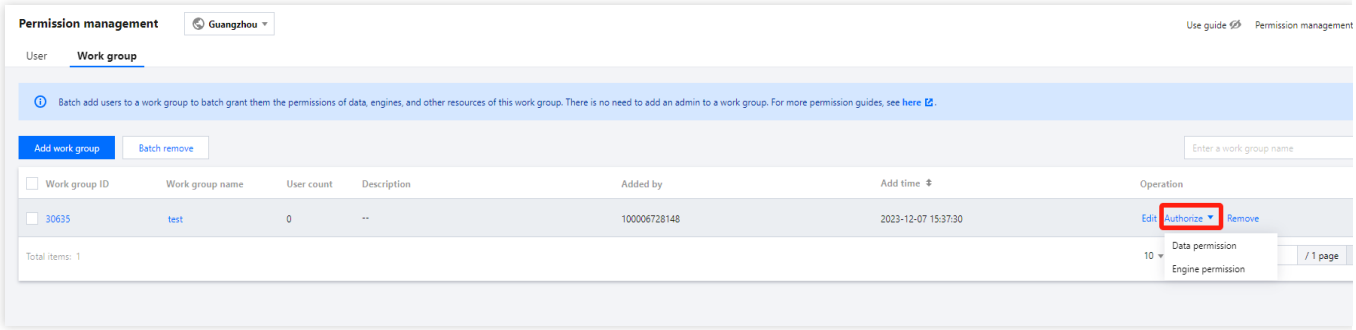


3. Associate a user: The associated user will acquire all permissions under the respective work group.



Granting permissions to a work group

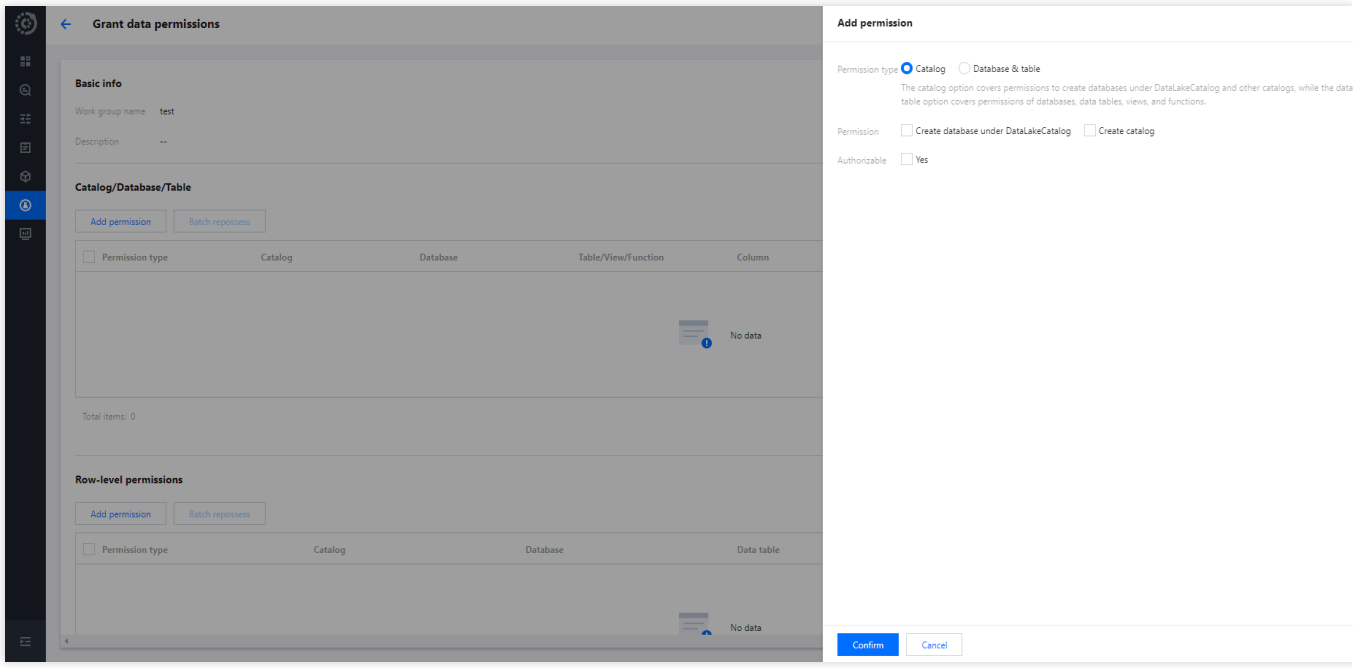
After creating the work group, click on the Authorize operation in the list to add permissions to the work group, including Data Permissions and Engine Permissions.



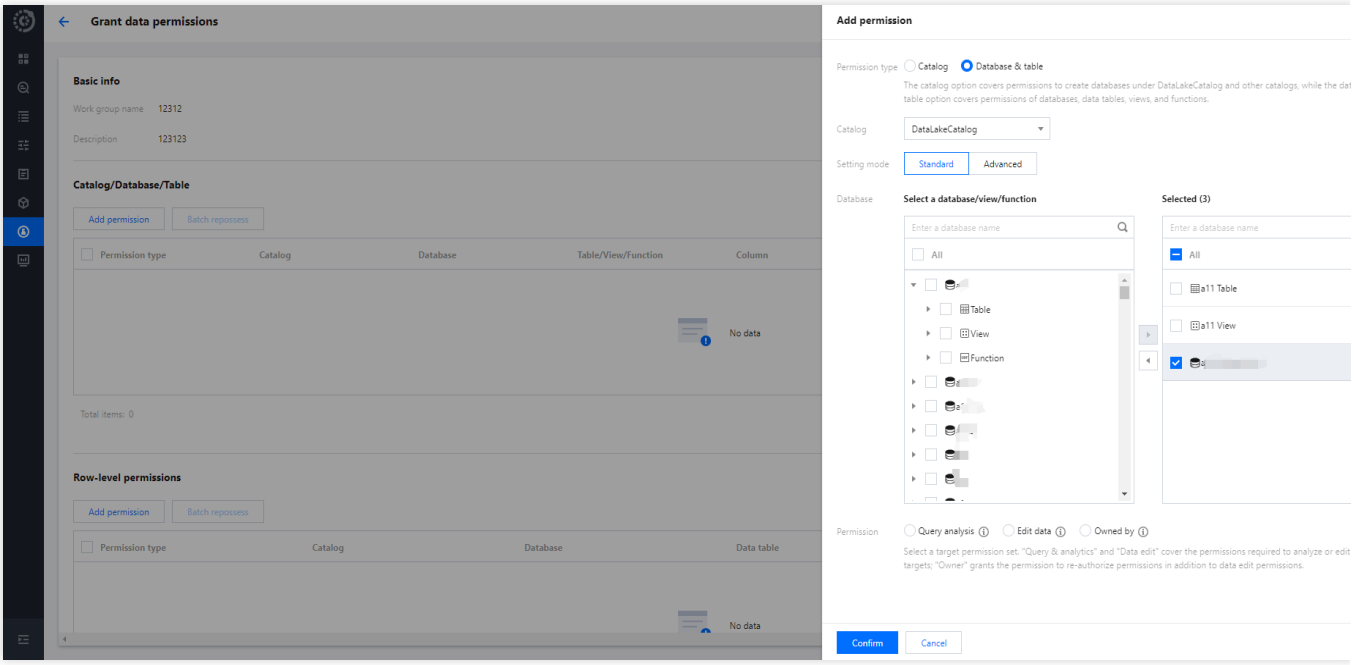
Data permission

Data permissions include:

Data Catalog Permissions: These include two types of permissions under the data catalog, namely, the ability to Create Database and Create Data Catalog.

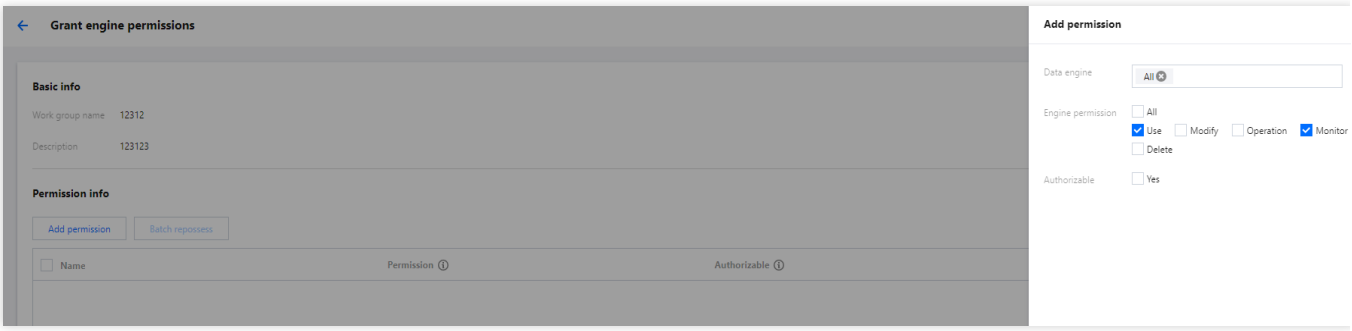


Database Table Permissions: Fine-grained permissions at the database table level can be granted, including query and edit permissions for databases, tables, views, and functions.



Engine permission

Select a data engine and grant the permissions to use, modify, or delete it.



Engine operation permissions are granted automatically

DLC supports default enablement of engine operation class permissions. Once enabled, all users will by default have the following permissions for that engine:

Utilize: Execute tasks using this engine.

Operation: Initiation of engine suspension or standby.

Monitoring: Administration of engine usage monitoring.

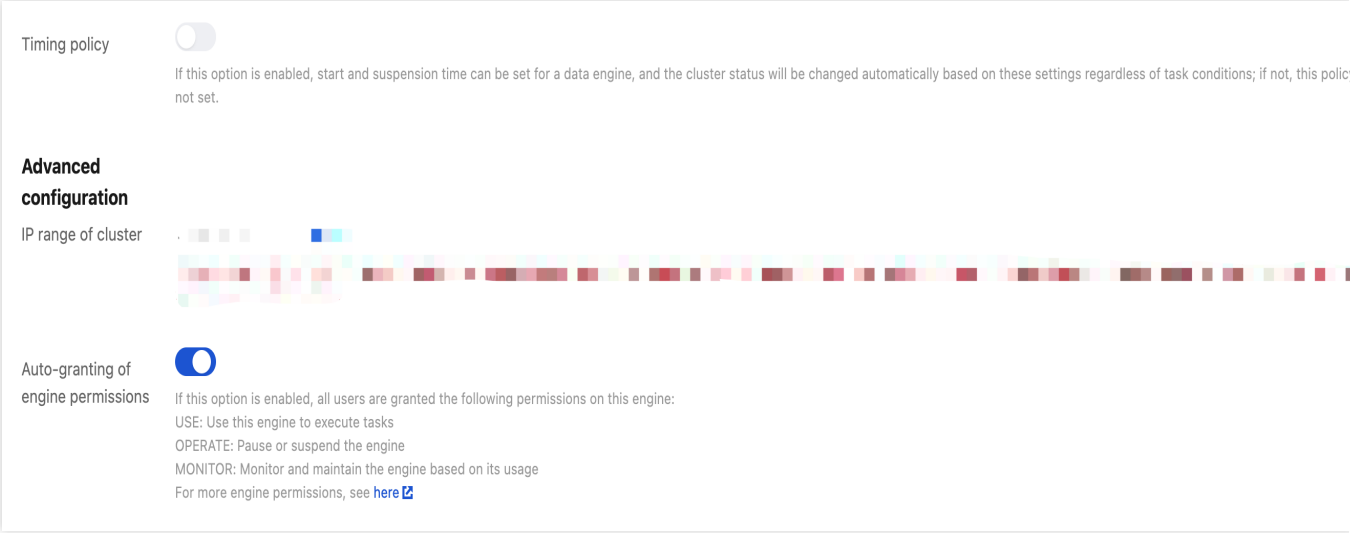
Note:

1. Upon termination, administrators inherently maintain all engine privileges. Ordinary users require an administrator to add permissions on the permission management page.
2. Existing ordinary user permissions will remain intact and can be deleted on the Permission Management page.
3. Subsequent newly created ordinary users have no usage rights, which should be manually added on the Permission Management page.

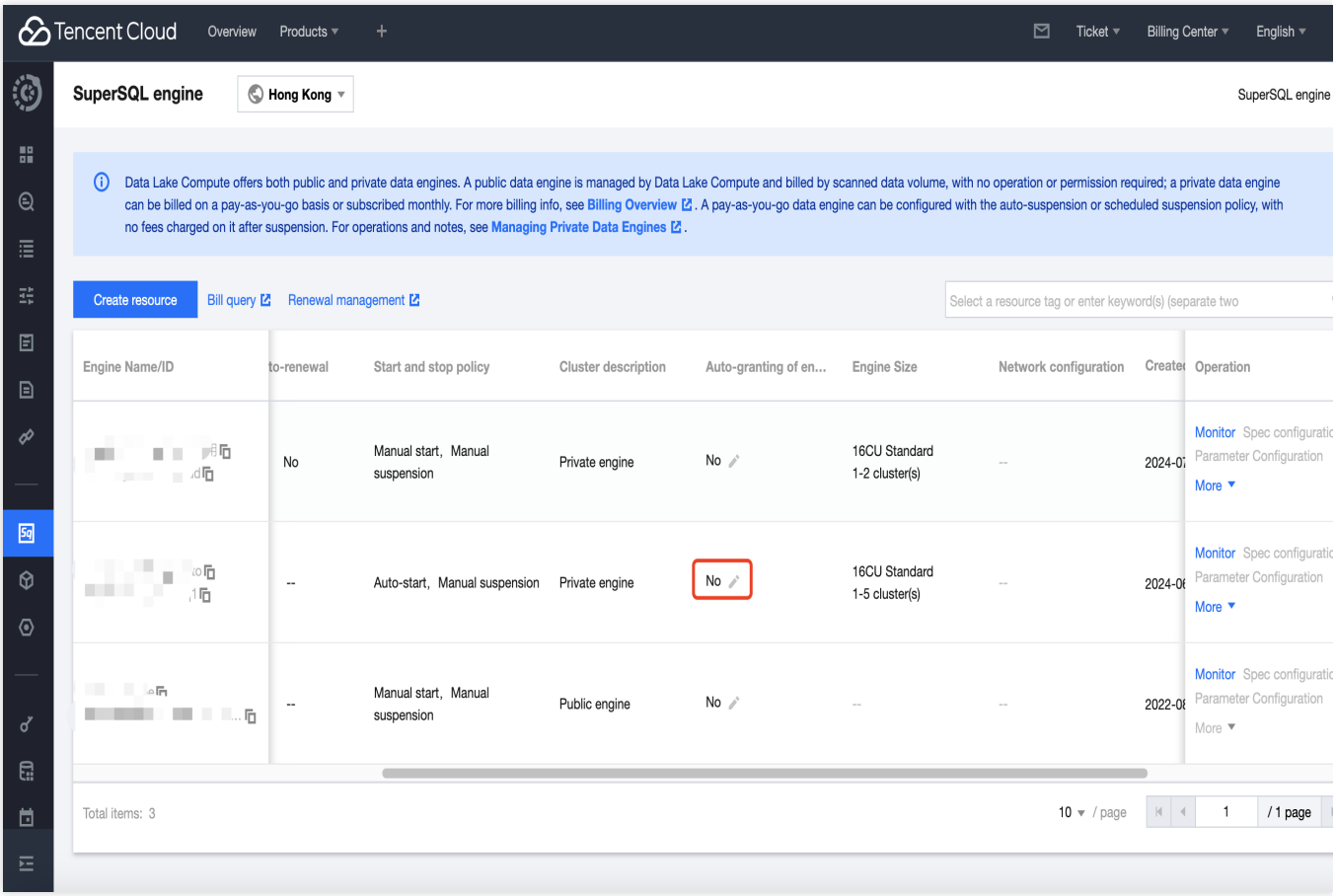
How do I enable or disable the self-delivery authorization engine

By default, the engine enables/disables two operation permission entries:

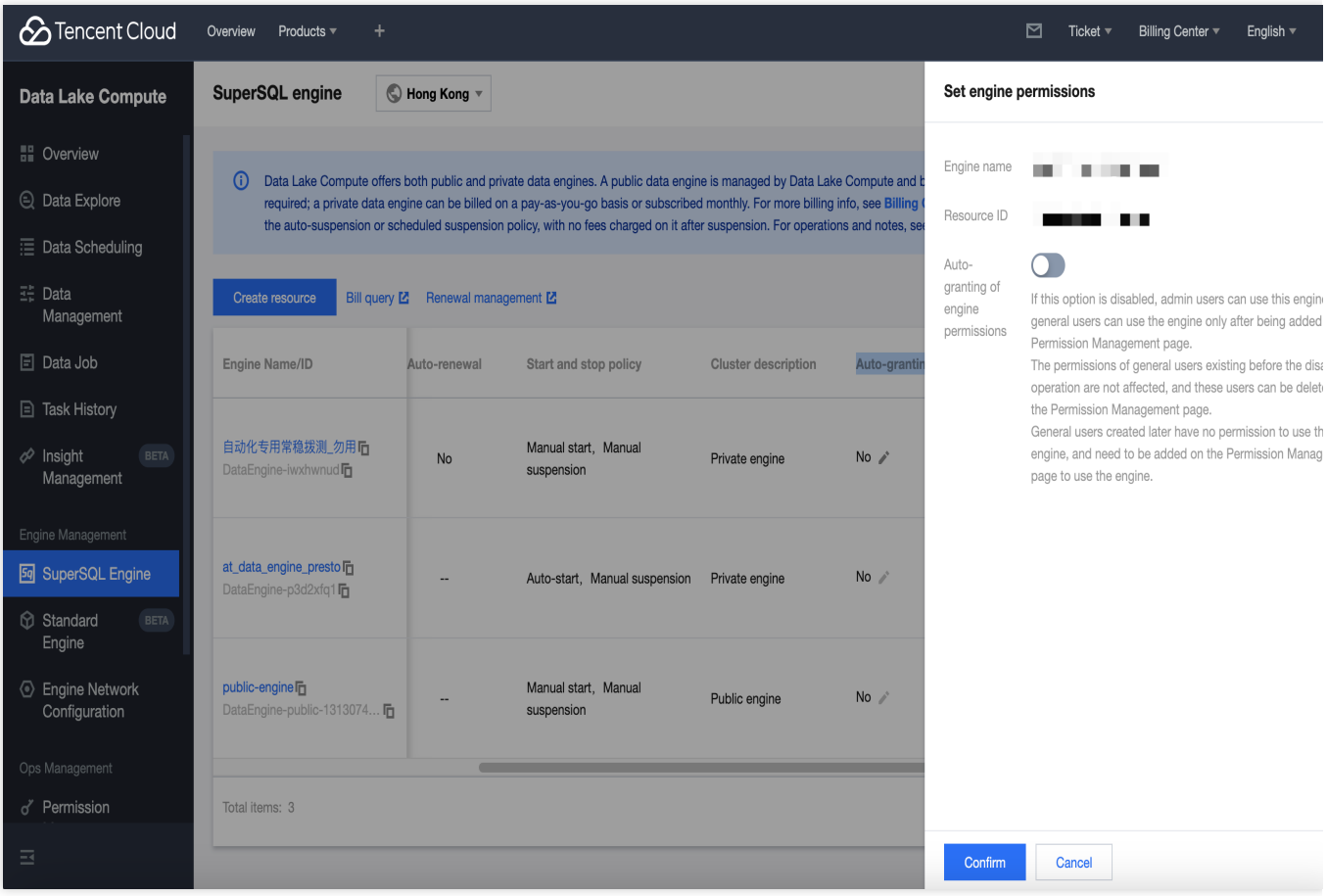
Access 1: Engine Purchase page > Advanced Configuration Items



Access 2: Go to the SuperSQL engine page and click Edit Auto-granting of engine permissions.



After setting engine permissions, click Confirm.


Tencent Cloud
Overview
Products
Ticket
Billing Center
English
Data Lake Compute
Overview
Data Explore
Data Scheduling
Data Management
Data Job
Task History
Insight Management
BETA
Engine Management
SuperSQL Engine
Standard Engine
BETA
Engine Network Configuration
Ops Management
Permission
SuperSQL engine
Hong Kong
Data Lake Compute offers both public and private data engines. A public data engine is managed by Data Lake Compute and b
required; a private data engine can be billed on a pay-as-you-go basis or subscribed monthly. For more billing info, see Billing
the auto-suspension or scheduled suspension policy, with no fees charged on it after suspension. For operations and notes, se
Create resource
Bill query
Renewal management
Engine Name/ID
Auto-renewal
Start and stop policy
Cluster description
Auto-grantin
自动化专用常稳拨测_勿用
DataEngine-iwxhwnud
No
Manual start, Manual suspension
Private engine
No
at_data_engine_presto
DataEngine-p3d2xfq1
--
Auto-start, Manual suspension
Private engine
No
public-engine
DataEngine-public-1313074...
--
Manual start, Manual suspension
Public engine
No
Total items: 3
Set engine permissions
Engine name
Resource ID
Auto-granting of engine permissions
If this option is disabled, admin users can use this engin
general users can use the engine only after being added
Permission Management page.
The permissions of general users existing before the dis
operation are not affected, and these users can be delet
the Permission Management page.
General users created later have no permission to use th
engine, and need to be added on the Permission Manag
page to use the engine.
Confirm
Cancel

Quick Start with Partition Table

Last updated：2024-07-17 15:25:14

Data Lake Compute Partition Table

With the partition catalog feature, you can store data with different characteristics in different catalogs. In this way, when exploring data, you can filter data by partition through the `where` condition. This greatly reduces the scanned data volume and improves the query efficiency.

Note:

Partitions in the same table should adopt the same data type and format.

Internal tables in Data Lake Compute are implemented as implicit partitions, so you don't need to care about the partition catalog structure.

Creating a Partition Table

Specify the partition field through the `PARTITIONED BY` parameter in the table creation statement.

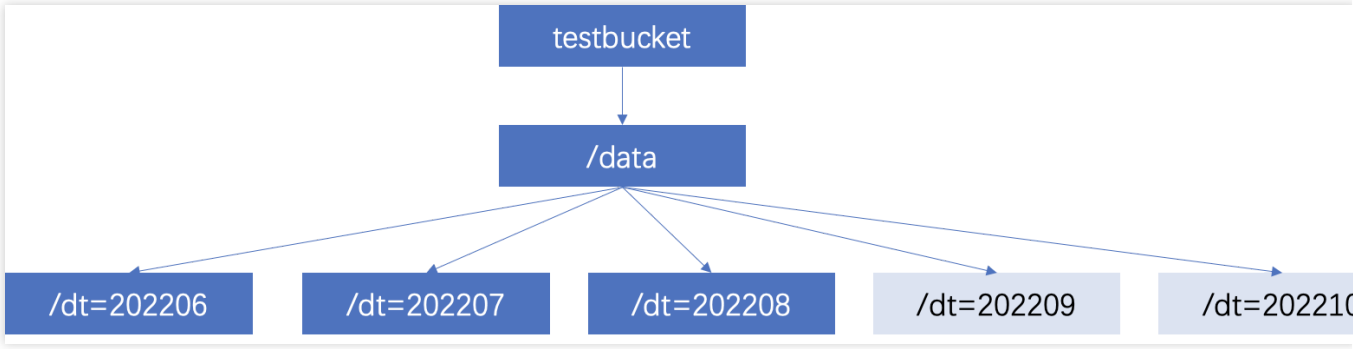
Example: Creating the `test_part` partition table

```
CREATE EXTERNAL TABLE IF NOT EXISTS `DataLakeCatalog`.`test_a_db`.`test_part` (
`_c0` int,
`_c1` int,
`_c2` string,
`dt` string
) USING PARQUET PARTITIONED BY (dt) LOCATION 'cosn://testbucket/data/';
```

Adding a Partition

Adding a partition through `ALTER TABLE ADD PARTITION`

If your data partition catalog uses the Hive partitioning rule (partition column name=partition column value), the rule can be used to add partitions. The catalog is organized as follows:




```
ALTER TABLE `DataLakeCatalog`.`test_a_db`.`test_part` add PARTITION (dt =
'202206')
ALTER TABLE `DataLakeCatalog`.`test_a_db`.`test_part` add PARTITION (dt =
'202207')
ALTER TABLE `DataLakeCatalog`.`test_a_db`.`test_part` add PARTITION (dt =
'202208')
ALTER TABLE `DataLakeCatalog`.`test_a_db`.`test_part` add PARTITION (dt =
'202209')
ALTER TABLE `DataLakeCatalog`.`test_a_db`.`test_part` add PARTITION (dt =
'202210')
```

Adding a partition by specifying the location through `ALTER TABLE`

If your data adopts a general COS catalog (not in the "partition column name=partition column value" format), you can specify a catalog when adding a partition.

Sample SQL:

```
ALTER TABLE `DataLakeCatalog`.`test_a_db`.`test_part` add PARTITION (dt =
'202211') LOCATION='cosn://testbucket/data2/202211'
ALTER TABLE `DataLakeCatalog`.`test_a_db`.`test_part` add PARTITION (dt =
'202212') LOCATION='cosn://testbucket/data2/202212'
```

Automatically adding a partition through `MSCK REPAIR TABLE`

Use the `MSCK REPAIR TABLE` statement to scan the data catalog specified during table creation. If there is a new partition catalog, the system will automatically add the partitions to the metadata of the data table.

Sample SQL:

```
MSCK REPAIR TABLE `DataLakeCatalog`.`test_a_db`.`test_part`
```

We recommend you use `ALTER TABLE` to add a partition preferably, as automatic adding through `MSCK REPAIR TABLE` has the following restraints:

`MSCK REPAIR TABLE` only adds partitions to the metadata of the data table but does not delete them.

`MSCK REPAIR TABLE` is not recommended if the data volume is large, as it will scan all the data, which may cause a timeout.

If your partition catalog doesn't use the Hive partitioning rule (partition column name=partition column value), `MSCK REPAIR TABLE` cannot be used.

Enabling Data Optimization

Last updated：2024-07-31 17:23:30

In big data scenarios, frequent fragmented writes generate a large number of small files, which significantly slow down performance. Based on extensive production practice experience, DLC offers you efficient, simple, and flexible data optimization capabilities that can handle near real-time scenarios with large data volumes.

Note:

1. In Upsert scenarios, a large number of small files and snapshots will be generated. You need to configure data optimization before writing to avoid the need for extensive resource processing of historical backlog of small files after writing.
2. Currently, data optimization capability only supports DLC native tables.
3. The initial execution of data optimization tasks may be slow, depending on the stock data volume size and the selected engine resource specifications.
4. It is recommended to separate the data optimization engine from the business engine to avoid the situation where data optimization tasks and business tasks compete for resources, causing delays in business tasks.

Configure data optimization through the DLC console

DLC data optimization strategies can also be set in the data directory, database, and data table. When data optimization strategies are not specifically set for a database or data table, they will inherit the optimization strategy from the previous level. When configuring data optimization, users need to select an engine. To execute data optimization tasks, if the user currently does not have a data engine, they may refer to Purchasing Dedicated Data Engine to make a purchase. DLC data governance supports Spark SQL Engine and Spark Job Engine.

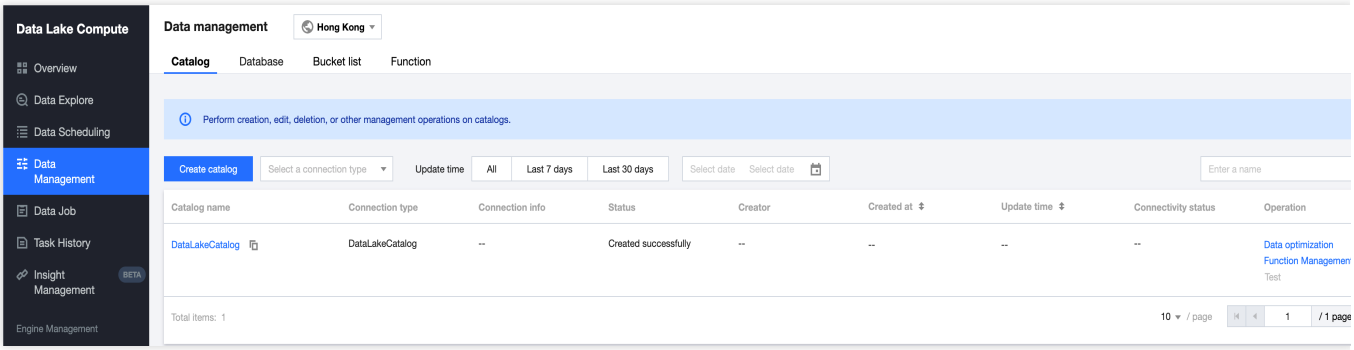
Note:

1. If a user chooses the Spark Job Engine as the data optimization resource, DLC will create data optimization tasks on that engine. Depending on the size of the cluster, the optimized task data created will vary. For instance, if the cluster size is smaller than 32 CU, one data optimization task will be created to execute all optimization tasks. If the cluster size is larger than 32 CU, two data optimization tasks will be created to separately execute write optimization and data deletion optimization.
2. When choosing a Spark Job as a data optimization resource, some resources need to be reserved. If the optimization tasks queue exceeds 50, DLC will launch temporary data optimization tasks to quickly process the backlog of optimization tasks.

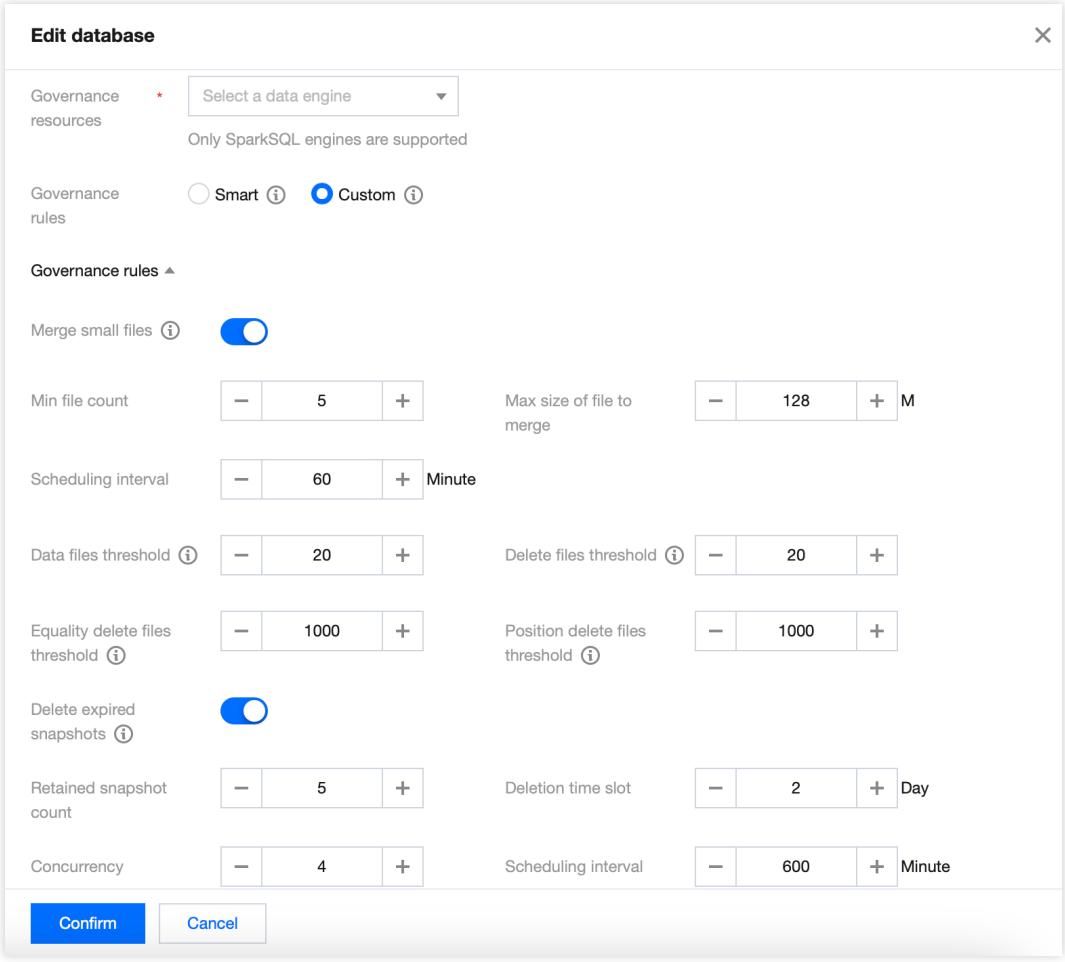
Data Directory Configuration Steps

You can use DLC's Data Catalog Editing Feature to configure data optimization capabilities for your data directory.

1. Go to the Data Management Module in the DLC Console, enter the **Data Management** page, and click **Data Optimization**.



2. Open the **Data Optimization** page of the data directory, configure the corresponding data optimization resources and policies. Once confirmed, the data optimization feature will automatically apply to that data directory.



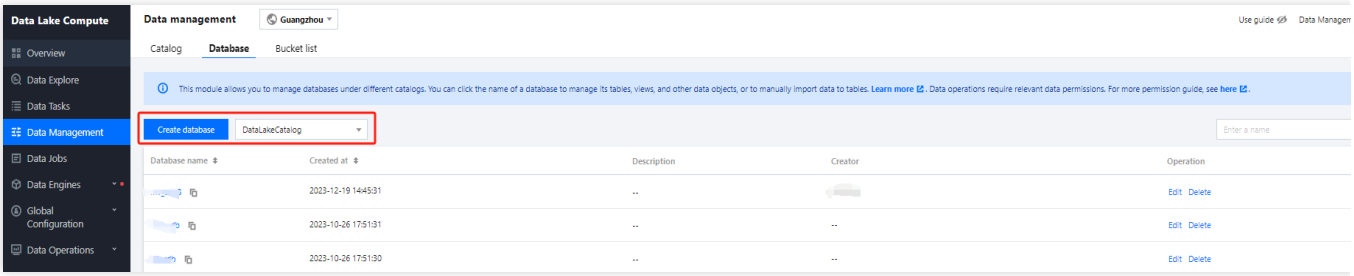
Note:

Only supports configuring data optimization for the DataLakeCatalog data directory.

Database Configuration Steps

If you want to configure a data optimization strategy for a specific database individually, you can use the database editing capabilities of DLC to configure data optimization capabilities for the database.

1. Enter the DLC console Data Management Module, enter the **Database** page, enter the database list under DataLakeCatalog.



2. Open the database page, click **Data Optimization Configuration**. Once confirmed, the data optimization strategy will automatically apply to that database.

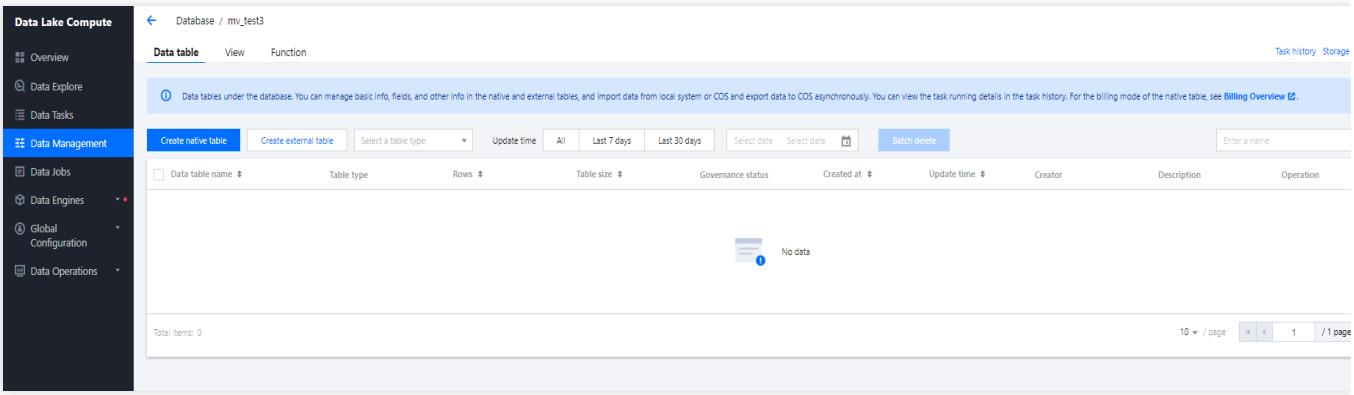
Note:

When creating a database and editing data, the default to show data optimization strategy inherits the data optimization strategy of the superior data directory. If you want to customize the data optimization strategy, you need to select **Custom Configuration** and configure data optimization resources and policies.

Data Table Configuration Steps

If you want to configure a data optimization strategy for a specific data table individually, you can use the data table editing capabilities of DLC to configure data optimization capabilities for the data table.

1. Enter the DLC console Data Management Module, enter the **Database** page, select a database, then enter the **Data Table** list page, and click **Create Native Table**.



2. Open the Create Native Table page, configure the corresponding optimization resources, and once confirmed, the data optimization strategy will automatically apply to that data table.
3. For already created tables, you can click **Data Optimization Configuration** to edit the existing data table's data optimization strategy.

Note:

When creating or editing a data table, the default data optimization strategy displayed inherits from the parent data table's data optimization strategy. If you want to customize the data optimization strategy, you need to select **Custom Configuration** and configure data optimization resources and policies.

Optimize data through attribute field configuration

Besides the above visualization method for configuring data optimization, you can also manually specify library and table field attributes for configuration. For example:

```
// for table govern policy
ALTER TABLE
    `DataLakeCatalog`.`wd_db`.`wd_tb`
SET
TBLPROPERTIES (
    'smart-optimizer.inherit' = 'none',
    'smart-optimizer.written.enable' = 'enable'
)
// for database govern policy
ALTER DATABASE
    `DataLakeCatalog`.`wd_db`
SET
DBPROPERTIES (
    'smart-optimizer.inherit' = 'none',
    'smart-optimizer.written.enable' = 'enable'
)
```

The attribute values for data optimization can be modified via the ALTER statement. The attribute value definitions are as follows:

Attribute Value	Meaning	Default Value	Value Description
smart-optimizer.inherit	Whether to Inherit from the Parent Strategy	default	none: Does not inherit default: Inherit
smart-optimizer.written.enable	Whether Write Optimization is Enabled	disable	disable: Not Enabled enable: Enabled

smart-optimizer.written.advance.compact-enable	(Optional) Advanced Write Optimization Parameters, Whether to Start Small File Merge	enable	disable: Not Enabled enable: Enabled
smart-optimizer.written.advance.delete-enable	(Optional) Advanced Write Optimization Parameters, Whether to Start Data Cleanup	enable	disable: Not Enabled enable: Enabled
smart-optimizer.written.advance.min-input-files	(Optional) Merge Minimum Number of Input Files	5	When the number of files in a table or partition exceeds the minimum number of files, the platform will automatically check and initiate file optimization merge. File optimization merge can effectively improve analyze query performance. The larger the minimum number of files, the higher the resource load. The smaller the minimum number of files, the more flexible the execution, and tasks will be more frequent. It is recommended to set the value to 5.
smart-optimizer.written.advance.target-file-size-bytes	(Optional) Merge Target Size	134217728 (128 MB)	During file optimization merge, files will be combined to meet the target size as much as possible. It is recommended to set the value to 128M.
smart-optimizer.written.advance.retain-last	(Optional) Snapshot Expiration Time, Unit Days	5	When the snapshot retention time exceeds this value, the platform will mark the snapshot as expired. The longer the snapshot expiration time, the slower the snapshot cleanup speed, and the more storage space is occupied.

smart-optimizer.written.advance.before-days	(Optional) Number of Expired Snapshots to Retain	2	Expired snapshots exceeding the retention count will be cleaned up. The more expired snapshots retained, the more storage space is occupied. It is recommended to set the value to 5.
smart-optimizer.written.advance.expired-snapshots-interval-min	(Optional) Snapshot Expiration Execution Cycle	600(10 hour)	The platform will periodically scan snapshots and expire them. The shorter the execution cycle, the more sensitive the snapshot expiration will be, but it may consume more resources.
smart-optimizer.written.advance.cow-compact-enable	(Optional) Enable Merge for COW Tables (V1 Table or V2 Non-Upsert Table)	disable	Once this configuration item is enabled, the system will automatically generate file merge tasks for COW tables. Note: COW tables usually have a large data volume, and file merging may consume a lot of resources. You can choose whether to enable file merging for COW tables based on resource availability and table size.
smart-optimizer.written.advance.strategy	(Optional) File Merge Strategy	binpack	binpack (default merge strategy): Merges data files that meet the merge conditions into larger data files using the append method. sort: The sort strategy merges files based on specified fields. You can choose query condition fields that are frequently used in your business scenarios as the sorting fields. Merging in this way can improve query performance.
smart-optimizer.written.advance.sort-order	(Optional) When the file merge strategy is sort, the configured sort collation	-	If you haven't configured a sorting strategy, the Upsert Table will sort using the configured upsert key values (by default, the first two key values) in an ASC NULLS LAST manner. If a sorting strategy cannot be found for COW Table during a sort merge,

			the binpack default merge strategy will be used.
smart-optimizer.written.advance.remove-orphan-interval-min	(Optional) Period for Removing Orphan Files	1440(24 hour)	The platform will periodically scan and clean up orphan files. The shorter the execution cycle, the more sensitive the cleanup of orphan files will be, but it may consume more resources.

Optimization Suggestions

The DLC backend regularly statistics native table metric items and combine these metrics with best practices to provide optimization suggestions for native tables. There are four categories of optimization suggestion items, including basic configuration for table usage scenarios, data optimization recommendations, and recommendations for data storage distribution items.

Optimization recommendation check items	Sub-check item	Meaning	Business Scenario	Optimization Suggestions
Basic attribute configuration check of the table	Metadata governance enabled	Check whether metadata governance is enabled to prevent metadata volume expansion due to frequent table writes	append/merger into/upsert	Recommended to enable
	Bloom filter set	Check if the bloom filter is set. After enabling the bloom filter for MOR tables, it quickly filters the deletes files, speeding up MOR table queries and deletes file merges	upsert	Must enable
	Metrics key attributes configured	Check if metrics are set to full. Once this attribute is enabled, it will record all metrics information, preventing incomplete metrics information recording due to excessively long table locations	append/merger into/upsert	Must enable

Data optimization configuration check	Small File Merge	Check if small file merging is enabled	merge into/upsert	Must enable
	Snapshot Expiration	Check if snapshot expiration is enabled	append/merge into/upsert	Recommended to enable
	Remove orphaned files	Check if removing orphaned files is enabled	append/merge into/upsert	Recommended to enable
Recent governance task check items	Recent governance task check items	If data governance is enabled, the system will track the execution of data governance tasks. If multiple tasks in a row time out or fail, it will be deemed in need of optimization	append/merger into/upsert	Recommended to enable
Data Storage Distribution	Average File Size	Collect summary information from snapshots, calculate the average file size, and if the average file size is less than 10MB, it will be deemed in need of optimization	append/merger into/upsert	Recommended to enable
	MetaData Meta File Size	Collect table metadata.json Meta File Size, if the file size exceeds 10MB, it will be deemed in need of optimization	append/merger into/upsert	Recommended to enable
	Number of Table Snapshots	Collect Number of Table Snapshots, if the number of snapshots exceeds 1000, it will be deemed in need of optimization	append/merger into/upsert	Recommended to enable

Optimization Suggestions for Basic Configuration Items of Table Attributes

Check and configure Metadata Governance Method

Step1 Inspection Method

Use 'show TBLPROPERTIES' to view table attributes and check if "write.metadata.delete-after-commit.enabled", "write.metadata.previous-versions-max" are configured.

Step2 Configuration Method

If Step1 finds that it's not configured, you can configure it using the following Alter table DDL, with the method referenced below.

```
ALTER TABLE
  `DataLakeCatalog`.`axitest`.`upsert_case`
SET
  TBLPROPERTIES(
    'write.metadata.delete-after-commit.enabled' = 'true',
    'write.metadata.previous-versions-max' = '100'
  );
```

Note:

To enable automatic metadata governance, "write.metadata.delete-after-commit.enabled" should be set to true. The number of historical metadata to retain can be set according to the actual situation, for example, setting "write.metadata.previous-versions-max" to 100 will retain up to 100 historical metadata.

Inspecting and Setting Bloom Filter Method

Step1 Inspection Method

Use show TBLPROPERTIES to view table attributes, and check if "write.parquet.bloom-filter-enabled.column.{column}" is set to true.

Step2 Configuration Method

If Step1 finds that it's not configured, you can configure it using the following Alter table DDL, with the method referenced below.

```
ALTER TABLE
  `DataLakeCatalog`.`axitest`.`upsert_case`
SET
  TBLPROPERTIES(
    'write.parquet.bloom-filter-enabled.column.id' = 'true'
  );
```

Note:

It is recommended to enable bloom in upsert scenarios, and configure it based on the upsert primary key. If there are multiple primary keys, it is advisable to set it for the first two primary key fields.

After updating the bloom fields, if there are upstream writes from inlong/oceans/flink, you must restart the upstream import job.

Check and configure table key attributes metrics

Step1 Inspection Method

View table properties using `show TBLPROPERTIES` and check if "write.metadata.metrics.default" is configured as "full".

Step2 Configuration Method

If Step1 finds that it's not configured, you can configure it using the following Alter table DDL, with the method referenced below.

```
ALTER TABLE
  `DataLakeCatalog`.`axitest`.`upsert_case`
SET
  TBLPROPERTIES('write.metadata.metrics.default' = 'full');
```

Data Optimization Configuration Recommendations

Step1 Inspection Method

Check using SQL

View table properties using `show TBLPROPERTIES` and check if data optimization is configured. Refer to DLC Native Table Core Capabilities for the attribute configuration values for data optimization.

Visual inspection through the DLC Console

Go to the Data Management Module in the DLC Console, enter the **Database** page, select a database to access the **Data Table** list page, choose the table to inspect, and proceed to **Data Optimization Configuration**.

Step2 Configuration Method

Follow the guidance to enable data optimization.

Recent recommendations for data governance optimization task items

Check if data governance is functioning properly

Step1 Inspection Method

Enter the DLC Console Data Management Module, enter the **Database** page, select a database and then enter the **Data Table** list page, click on the data table name, enter **Optimized Monitoring**, choose **Optimization Task** then select **Today's Optimization**, check for tasks that failed in the last three hours, if there are any, the check is not passed. Select the failed task, in **View Details** look at the **Execution Results**.

Step2 Fix Methods

Summary of Reasons and Solutions for Failed Scenario Data Optimization Tasks.

1. Data Governance Configuration Error led to failure.

Sort Merge Strategy was enabled, but the collation was incorrectly configured, or a nonexistent rule was set.

The configuration for the data governance engine has changed, leading to the inability to find an appropriate engine when running governance tasks.

2. Task Execution Timed Out.

Note:

After repairing the recent data optimization task performance, it is necessary to wait three hours before checking if it has recovered.

Data Storage Distribution Item Optimization Suggestions

Note:

Failure in this scenario check is usually due to large data volume. It's recommended to handle it manually before considering addition to Data Optimization Governance.

It is recommended to use the more efficient Spark job engine.

When manually merging small files, configure the target-file-size-bytes parameter based on the business scenario. For upsert operations, it is advised not to exceed 134217728, i.e., 128M. For append/merge into operations, it is advised not to exceed 536870912, i.e., 512M.

When using the Spark job engine to handle snapshot expiration, the max_concurrent_deletes parameter can be increased.

Average Data File Size Check Failure Handling Method

Step1 Summary of Reasons

The average size of data files is too small, usually occurring in the following scenarios:

The table is partitioned too finely, resulting in each partition having only a small amount of data.

When tables are written using the Insert into/overwrite method, the upstream data is dispersed, such as when the upstream data is also from a partitioned table with little data within partitions.

The table is written to the MOR Table using the merge into method, but small file merging has not been performed.

The table is written using the upsert method, but small file merging has not been performed.

Step2 Fix Methods

Refer to the following SQL to manually perform small file merging.

```
CALL `DataLakeCatalog`.`system`.`rewrite_data_files`(
  `table` => 'test_db.test_tb',
  `options` => map(
    'delete-file-threshold',
    '10',
    'max-concurrent-file-group-rewrites',  --Subject to actual resource conditions,
    '5',
    'partial-progress.enabled',
    'true',
    'partial-progress.max-commits',
    '10',
    'max-file-group-size-bytes',
    '10737418240',
    'min-input-files',
    '30',
    'target-file-size-bytes',
    '134217728'
```

```
    )
)
```

MetaData Meta File Size Check Failure Handling Method

Step1 Summary of Reasons

MetaData file size is too large, usually caused by an excessive number of data files, mainly due to the following reasons:

The table has been written to using the append method for a long time, and each write involves a large number of scattered files.

The table has the attributes of an MOR table and has been written to long-term using the merge into method, but small file merging is not enabled.

The table has not undergone snapshot expiration for an extended period, maintaining a large number of historical snapshot data files.

The table partitions are large, and each partition contains a large number of small files.

Step2 Fix Methods

Refer to manually perform small file merging.

Refer to the following SQL to manually execute the expired snapshot SQL and clean up historical snapshots.

```
CALL DataLakeCatalog.system.rewrite_data_files(
  table => 'test_db.test_tb',
  options => map(
    'delete-file-threshold',
    '10',
    'max-concurrent-file-group-rewrites', --The higher the concurrency,  and the fa
    '5',
    'partial-progress.enabled',
    'true',
    'partial-progress.max-commits',
    '10',
    'max-file-group-size-bytes',
    '10737418240',
    'min-input-files',
    '30',
    'target-file-size-bytes',
    '134217728'
  )
)
```

Based on the service scenario, the written files are aggregated to a certain extent to avoid scattered files.

If the data is written into insert into/insert overwrite, you can automatically add a repartition in either of the following ways.

1. This parameter takes effect when both of the following parameters are true. In this case, you can use the preceding parameters to control the number or size of automatically adapted partitions after repartition.
spark.sql.adaptive.enabled：This parameter must be true. The default value is true for cluster creation.
spark.sql.adaptive.insert.repartition：This parameter must be true. The default value is false for cluster creation.
2. Specify the following parameters to take effect. This case repartition spark. The partition number after SQL. The adaptive. Insert. The repartition. ForceNum the specified value.
spark.sql.adaptive.insert.repartition.forceNum：This parameter specifies the value of the partition to be partitioned. It is left blank by default when the cluster is created.

Check the number of snapshots. This operation fails to pass the check

Step1 Cause summary

Snapshots do not expire for a long time.
The upsert writes data to the checkpoint interval improperly, resulting in a large number of snapshots.

Step2 Repair method

See Snapshot expiration SQL to perform snapshot expiration operations.
Adjust the flink write checkpoint interval. It is recommended that the checkpoint interval of DLC native table upsert be 3 to 5 minutes.

Cross-Source Analysis of EMR Hive Data

Last updated：2024-07-17 15:27:21

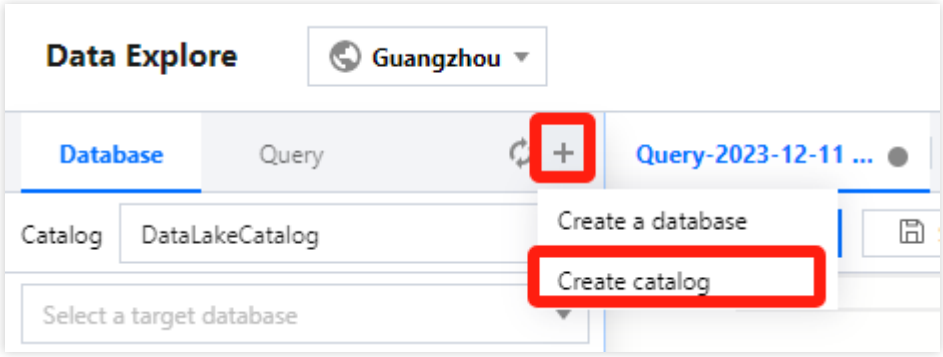
Data Lake Compute allows you to configure an EMR Hive data source for multi-source federated data analysis.

Preparations

Get the EMR Hive address.
Use an account with the permission to create data catalogs. For more information on permissions, see Permission Overview.

Creating an EMR Hive data source

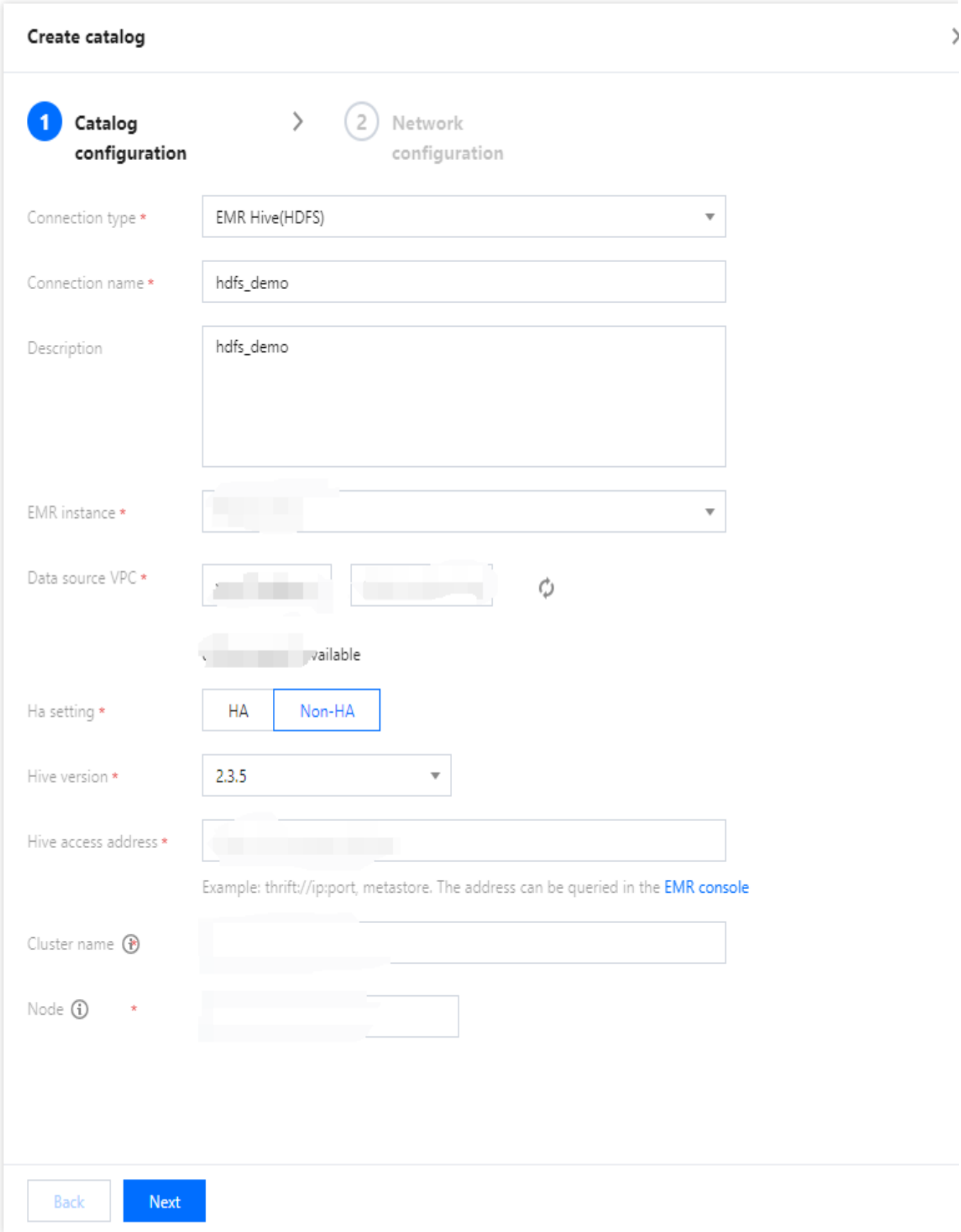
1. Log in to the Data Lake Compute console and select the service region.
2. Select **Data Explore** on the left sidebar, click **+** in the **Database & table** column, and select **Create data catalog**.



3. Select **EMR Hive (HDFS)** for **Connection type** and select the target EMR instance. The VPC information will be populated by default after the instance is selected. **EMR versions supported by EMR Hive are 2.3.5, 2.3.7, 3.1.1, and 3.1.2.**

Note:

Relevant permissions are required for you to select the EMR Hive instance.



4. Select the **Run cluster**. Currently, you can only select a private data engine of Presto. If there is no engine, create one on the **Data engine** page. For more information on the purchase process, see Purchasing Private Data Engine.

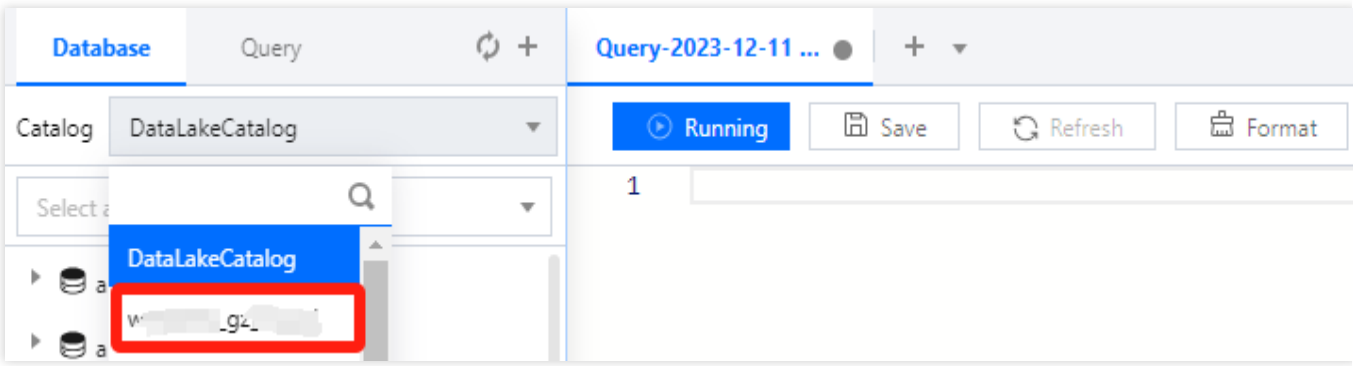
Note:

The IP range of the selected data engine cannot be the same as that of the EMR instance; otherwise, a network conflict will occur, and you cannot query or analyze data.

5. Click **Confirm**.

Querying the EMR Hive data

After the data catalog is created, you can switch to it from the **Data catalog** menu on the **Data Explore** page.

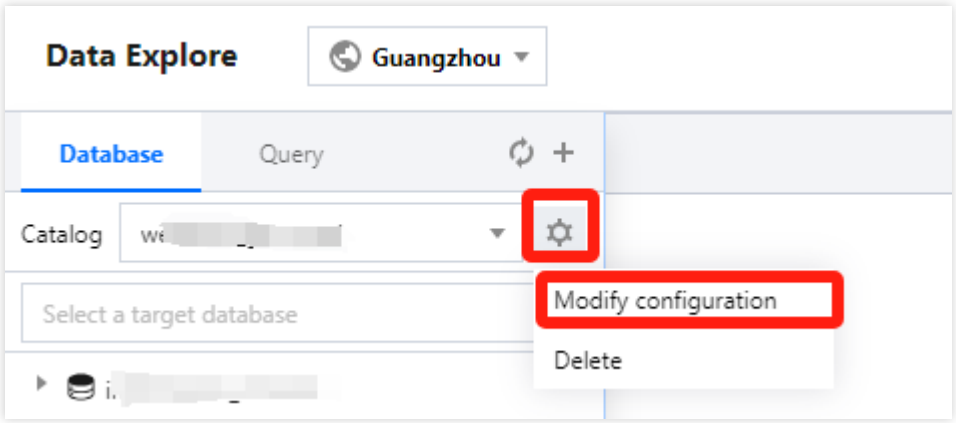


At this point, you can query and analyze the data catalog with SQL statements.

Select the data engine bound when the data catalog is created and click **Run** to get the query result.

Note:

You can only query the data catalog with its bound data engine. To change the bound engine, click the set icon next to the data catalog.



Standard Engine Configuration Guide

Last updated：2024-09-04 11:11:02

DLC offers two types of engines: the Standard Engine and the SuperSQL Engine. For a detailed comparison, see the table below or see the Data Engine Introduction. You can select the appropriate engine based on your specific business needs. **If you choose the Standard Engine, you can follow the instructions in this document for configuration and usage.**

Engine Types	Available Types	Main Features	Usage Requirements	Purchase Recommendations
Standard Engine	Spark Presto	Integrated Spark: The Standard Spark Engine supports native syntax from the Spark/Presto community, making it easy to learn and migrate. Flexible usage: Supports both Hive JDBC and Presto JDBC. Integrated Spark: The Standard Spark Engine can execute SQL and Spark batch tasks.	The free Gateway specification is 2 CU.	1. Requires the use of Spark/Presto native syntax. 2. Prefer to purchase a Spark engine for batch jobs and offline SQL tasks. 3. Prefer to use Hive JDBC and Presto JDBC.
SuperSQL Engine	SparkSQL\\nSpark Jobs\\nPresto	Unified syntax: A single syntax is applicable to both Spark and Presto engines. Supports federated queries.	Requires learning the SuperSQL unified syntax.\\nFor SQL/batch tasks, it is recommended to purchase the corresponding engine type.	1. Prefer to use Spark + Presto unified syntax. 2. Federated queries are required.

Note:

1. Before purchasing, you should ensure that your account has been granted financial permissions in CAM.

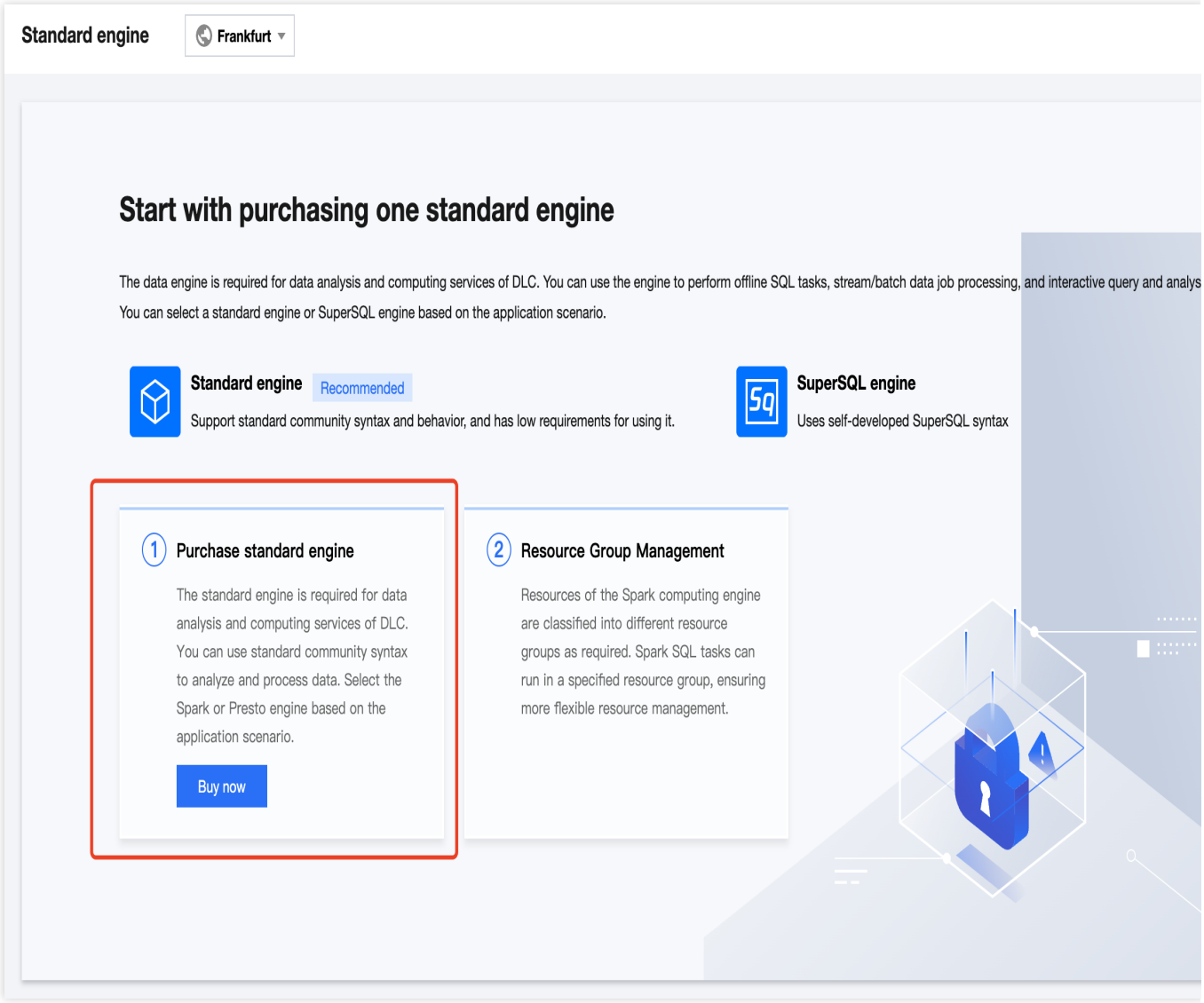
2. Resources cannot be used across regions, so confirm that the current region is correct before purchasing.

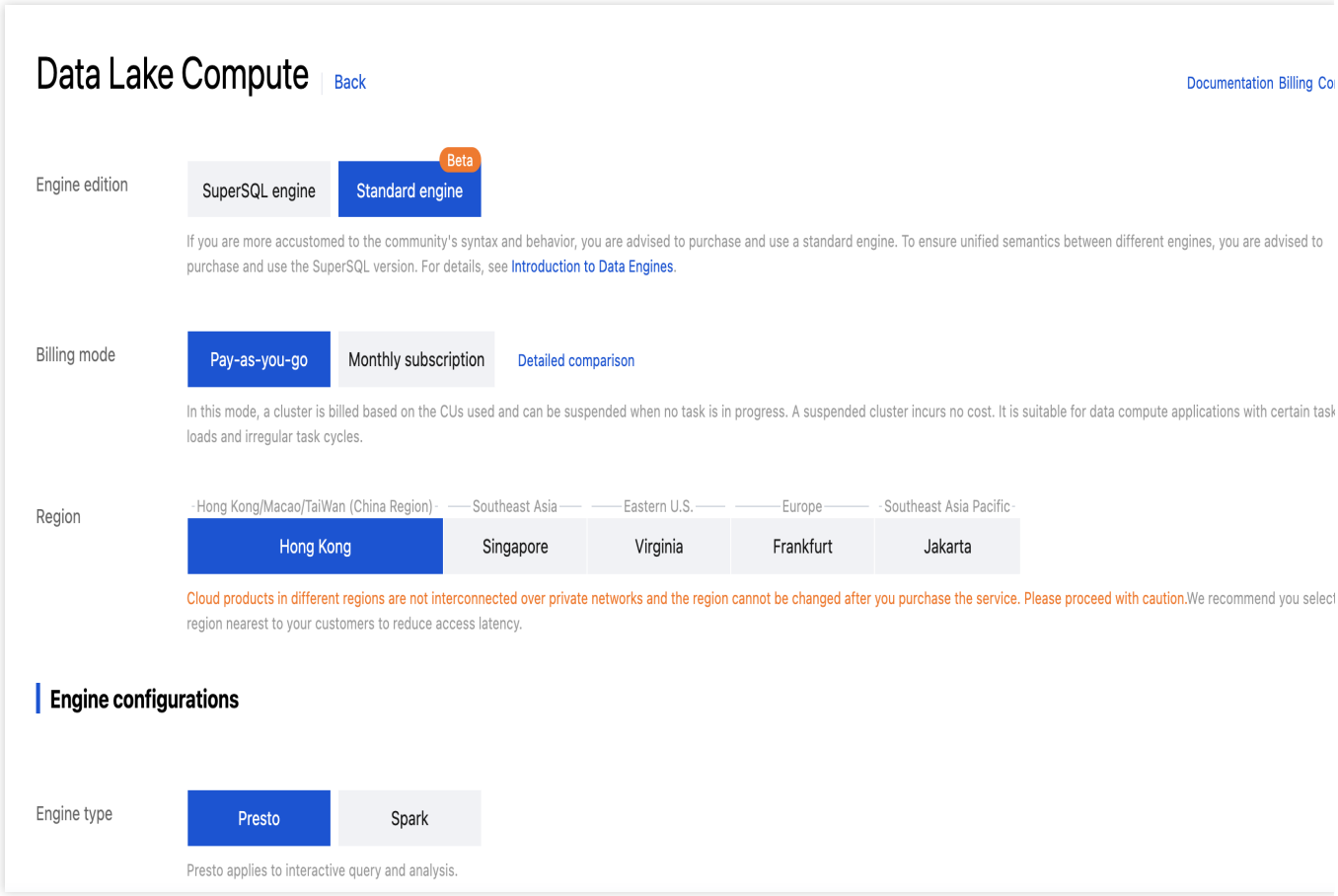
Standard Engine Configuration Guide

After completing the purchase and configuration of the Standard Engine, you can use it within DLC's **Data Exploration**. Additionally, for the Spark Standard Engine, if you have multi-tenant or task isolation requirements, you can also configure **Resource Group** for resource allocation and isolation. The detailed guide is as follows:

Step 1: Purchasing the Engine

Note:
1. Engines cannot be used across regions.
2. Engine specification recommendation: Since a 16 CU cluster is relatively small, it is recommended only for testing scenes. For real production environments, it is recommended to choose a cluster with a specification of 64 CUs or more.
3. Engine network configuration: Custom network configurations can be set during the initial purchase. If you need to make changes later, please Submit Ticket to apply for modifications.


Standard engine
Frankfurt
Start with purchasing one standard engine
The data engine is required for data analysis and computing services of DLC. You can use the engine to perform offline SQL tasks, stream/batch data job processing, and interactive query and analys
You can select a standard engine or SuperSQL engine based on the application scenario.
Standard engine
Recommended
Support standard community syntax and behavior, and has low requirements for using it.
SuperSQL engine
Uses self-developed SuperSQL syntax
1
Purchase standard engine
The standard engine is required for data analysis and computing services of DLC. You can use standard community syntax to analyze and process data. Select the Spark or Presto engine based on the application scenario.
Buy now
2
Resource Group Management
Resources of the Spark computing engine are classified into different resource groups as required. Spark SQL tasks can run in a specified resource group, ensuring more flexible resource management.



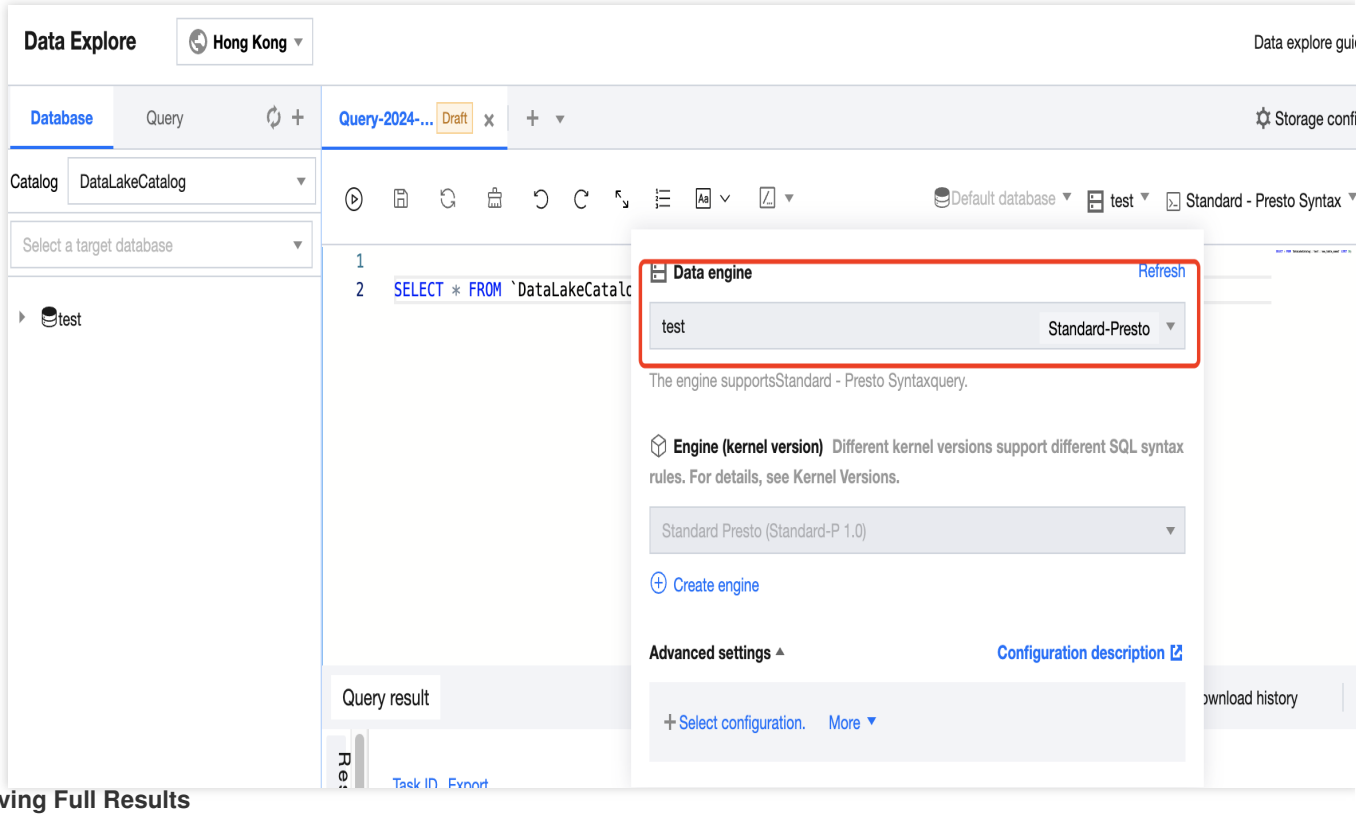
Step 2: Using Data Exploration

Selecting the Standard Engine for Queries

Note:

Depending on the type of Standard Engine, you may need to switch to the corresponding syntax for queries.

If you select the Standard Spark Engine in Data Exploration, you can allocate task resources by using the DLC default resource group, a created resource group, or a one-time resource group (custom configuration).



Retrieving Full Results

Currently, the Standard Engine only supports returning up to 1,000 query results in the console. To retrieve the full results, you can see the following methods:

Engine	Retrieval Method
Standard Spark Engine	1. Users can configure the engine to automatically save query results to a COS path or view them in DLC's managed storage. 2. Results can be downloaded locally for review.
Standard Presto Engine	Retrieve full results via JDBC.

Step 3: Configuring Resource Groups (Optional)

Resource groups provide a secondary queue division of computing resources within the Spark Standard Engine. For a detailed introduction, see Resource Group Introduction. The computing units (CUs) of the DLC Spark Standard Engine can be allocated across multiple resource groups as needed. You can set the minimum and maximum CU limits for each resource group, along with start/stop policies, concurrency levels, and dynamic/static parameters, ensuring resource isolation and efficient workload management in complex scenes such as multi-tenancy and multi-tasking.

When you purchase a Standard Spark Engine, DLC provides a default resource group and also allows you to create multiple custom resource groups based on your specific business needs for flexible usage.

Note:

An engine can have a one-to-many relationship with resource groups. For example, Engine A can have several resource groups.

Managing and Configuring Resource Groups

1. Click to enter the resource group management of the corresponding engine.
2. Enter the Resource Management Group interface, and click **Create Resource Group** to configure a custom resource group. Alternatively, you can view and use the DLC default-configured resource group (no configuration required).

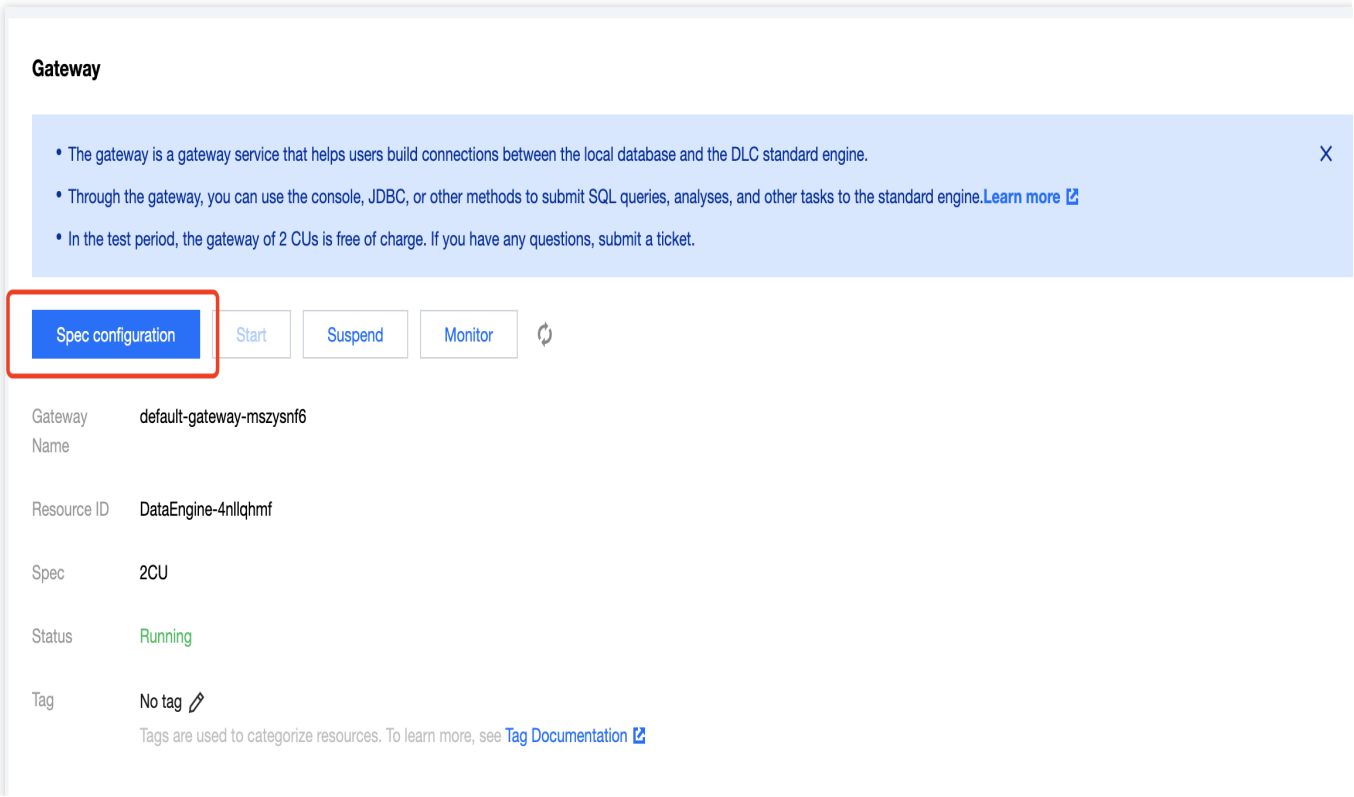
Appendix

Recommendations for Selecting Gateway Specifications

Gateway Specification	Spark Batch Instant Concurrency (Submitted/Running Tasks)	Concurrent Spark SQL/Presto SQL Queries	Number of Presto Engines Managed	Number of Spark Resource Groups Managed	Gateway HA
2 CU	30/50	100	4	50	No
16 CU	80/150	250	12	150	Yes
32 CU	220/400	600	35	400	Yes
64 CU	400/600	1000	70	700	Yes

Note:

The gateway is provided by default with a 2 CU specification (free of charge). If you need to upgrade the specifications, you can click Gateway details → select Specification Configuration to adjust and purchase.


Gateway
• The gateway is a gateway service that helps users build connections between the local database and the DLC standard engine.
• Through the gateway, you can use the console, JDBC, or other methods to submit SQL queries, analyses, and other tasks to the standard engine.Learn more
• In the test period, the gateway of 2 CUs is free of charge. If you have any questions, submit a ticket.
Spec configuration
Start
Suspend
Monitor
Gateway Name
default-gateway-mszysnf6
Resource ID
DataEngine-4nllqhmf
Spec
2CU
Status
Running
Tag
No tag
Tags are used to categorize resources. To learn more, see Tag Documentation