Data Warehouse Service (DWS)
8.1.3.333

AutoPilot User Guide

Issue	01
Date	2024-08-09

HUAWEI CLOUD COMPUTING TECHNOLOGIES CO., LTD.

Trademarks and Permissions


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

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

Contents

1 Introduction to AutoPilot

On the live network, due to different cluster scales and user services, the monitoring requirements of each user are different. With the running of services on the live network, the projects and cluster risks that need to be monitored may need to be changed at any time point.

To meet this requirement, GaussDB(DWS) provides a visualized monitoring framework. With this framework, users can create monitoring items and dashboards based on their O&M requirements. This greatly improves O&M flexibility. Monitoring metrics of key items can be brought online within a day rather than having to be created in a new GaussDB(DWS) version.

NOTICE

DMS 8.3.0 and later versions support AutoPilot. The DMS service needs to be switched from the DBM database to the IoT data warehouse. For details, see **Switching the DBM Database to the Stream Data Warehouse**.

2 Switching the DBM Database to the Stream Data Warehouse

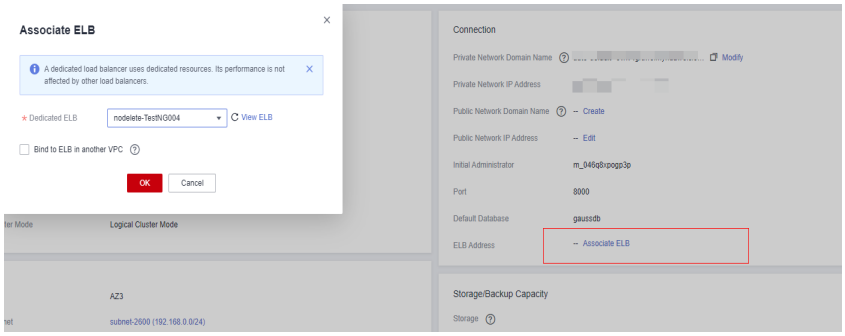
Step 1 Preparing the GaussDB(DWS) stream data warehouse

1. Creating a GaussDB(DWS) stream data warehouse

   Log in to the GaussDB(DWS) console and create a stream data warehouse with the following specifications. You are advised to create an independent VPC and tenant account for the DMS stream data warehouse, and then create a GaussDB(DWS) stream data warehouse in the same VPC.

   Preferred specifications: **dwsx2.rt.8xlarge.m6 | 96 vCPUs | 768 GB memory | 240,000 GB hot data storage**. The monitoring capability supports more than 1000 nodes.

2. Bind an ELB. For details, see section "Binding an ELB to a Cluster" and "Unbinding an ELB from a Cluster" in the *Data Warehouse Service User Guide*.



Step 2 Establishing network connection

Use Direct Connect to enable the network between the management plane and the tenant VPC where the GaussDB(DWS) stream data warehouse resides. For details, contact Direct Connect technical support.

Step 3 Migrating DMS Metadata

1. Log in to the DMS DBM database node and run the following command to export metadata to the **dms_meta_data.sql** file:

```
gs_dump  -p 8635 dms -f dms_meta_data.sql -a --column-inserts -t DMS_AGENT_TASK_RECORD -t DMS_DDL_EXAMINE_CLUSTER_INFO -t DMS_DDL_EXAMINE_COLLECTION_DATA -t DMS_DDL_EXAMINE_META_DATA -t DMS_META_AGEING -t DMS_META_AGENT_VERSION -t DMS_META_CLUSTER -t DMS_META_CLUSTER_ALARM_BIND_MAP -t
```



```
DMS_META_CLUSTER_ALARM_CONFIG_MAP  -t DMS_META_CLUSTER_ALARM_POLICY -t
DMS_META_CLUSTER_ALARM_RECORD -t DMS_META_CLUSTER_ALARM_RULE -t
DMS_META_COLLECTION_CONFIG -t DMS_META_HOST -t DMS_META_HOST_DISK -t
DMS_META_HOST_DISKFS -t DMS_META_HOST_NETIF -t DMS_META_INSTANCE -t
DMS_META_PERFORMANCE_ITEM -t DMS_META_PERFORMANCE_PANEL -t
DMS_META_WDR_CONFIG -t DMS_SQL_PROBE_INFO -t DMS_SQL_PROBE_RECORD -t
DMS_TASK_RECORD -t DMS_STATISTIC_STATUS -t DMS_STATISTIC_RESOURCE -t
DMS_STATISTIC_CLUSTER_LOAD
```

2. Download the gsql CLI client of the required version from the GaussDB(DWS) console. For details, see "Downloading a Client" in the *Data Warehouse Service (DWS) User Guide*. Log in to the **cdk-master** node, decompress the software package, and configure the following environment variables:

```
export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:/${client_path}/lib
export PATH=${PATH}:/${client_path}/bin
```

   NOTE

   ${client_path} indicates the directory where the gsql CLI client is located.

3. On the **cdk-master** node, run the following command to create a DMS metric database:

```
gsql -p 8000 -d gaussdb -U dbadmin -W ${pwd} -r -h ${cn_addr} -x 'create database dms'
```

   NOTE

   - **${pwd}** indicates the password of the stream data warehouse user **dbadmin**.
   - **${cn_addr}** indicates the database IP address.

4. Obtain the latest database initialization file **initial_dms_for_iot.sql** and run the following command on the **cdk-master** node to initialize the DMS monitoring metric table:

```
gsql -p 8000 -d dms -U dbadmin -W ${pwd} -r -h ${cn_addr} -f initial_dms_for_iot.sql
```

5. On the **cdk-master** node, use the gsql client to log in to the GaussDB(DWS) stream database and clear the initialization data.

```
truncate table DMS_DDL_EXAMINE_META_DATA;
truncate table dms_meta_ageing;
truncate table  dms_meta_collection_config;
truncate table  dms_meta_wdr_config;
truncate table  DMS_STATISTIC_STATUS;
truncate table  DMS_STATISTIC_RESOURCE;
truncate table  DMS_STATISTIC_CLUSTER_LOAD;
```

6. Run the following command on the **cdk-master** node to import the metadata **dms_meta_data.sql**:

```
gsql -p 8000 -d dms -U dbadmin -W ${pwd} -r -h ${cn_addr} -f dms_meta_data.sql
```

7. Initialize the **dms_mtc_max_ctime** table.

```
INSERT IGNORE INTO dms_mtc_max_ctime (VIRTUAL_CLUSTER_ID, CLUSTER_ID, TABLE_NAME,
host_id, host_name, MAX_CTIME, extra_condition)
with cluster as (SELECT VIRTUAL_CLUSTER_ID, CLUSTER_ID from DMS_META_CLUSTER where
cluster_status <> 'DELETED'),
host as (SELECT c.CLUSTER_ID,host_id, host_name from DMS_META_HOST h JOIN cluster c on
h.CLUSTER_ID = c.CLUSTER_ID),
collection as (SELECT c.VIRTUAL_CLUSTER_ID, c.CLUSTER_ID,
METRIC_TABLE_NAME,INIT_COLLECTION_TIME from DMS_META_COLLECTION_CONFIG cc JOIN
cluster c on cc.CLUSTER_ID = c.CLUSTER_ID where cc.PLUGIN_TYPE = 'metric' and
cc.COLLECTION_TYPE = 'HOST,ALL')
SELECT c.VIRTUAL_CLUSTER_ID, c.CLUSTER_ID, METRIC_TABLE_NAME, host_id,
host_name,INIT_COLLECTION_TIME * 1000, '-1' from host h JOIN collection c on h.CLUSTER_ID =
c.CLUSTER_ID;
INSERT IGNORE INTO dms_mtc_max_ctime (MAX_CTIME, VIRTUAL_CLUSTER_ID, CLUSTER_ID,
TABLE_NAME, inst_name,  extra_condition)
with cluster as (SELECT VIRTUAL_CLUSTER_ID, CLUSTER_ID from DMS_META_CLUSTER where
cluster_status <> 'DELETED'),
collection as (SELECT c.VIRTUAL_CLUSTER_ID, c.CLUSTER_ID,
METRIC_TABLE_NAME,INIT_COLLECTION_TIME from DMS_META_COLLECTION_CONFIG cc JOIN
```



```
cluster c on cc.CLUSTER_ID = c.CLUSTER_ID where cc.PLUGIN_TYPE = 'metric' and
cc.COLLECTION_TYPE like '%ANY%')
SELECT INIT_COLLECTION_TIME * 1000, VIRTUAL_CLUSTER_ID, CLUSTER_ID, METRIC_TABLE_NAME,
'default_inst_name', '-1' from collection;
INSERT IGNORE INTO dms_mtc_max_ctime (MAX_CTIME, VIRTUAL_CLUSTER_ID, CLUSTER_ID,
TABLE_NAME, INST_NAME, HOST_ID, INST_ID,  extra_condition)
with cluster as (SELECT VIRTUAL_CLUSTER_ID, CLUSTER_ID from DMS_META_CLUSTER where
cluster_status <> 'DELETED'),
inst as (SELECT c.CLUSTER_ID,INST_NAME, HOST_ID, INST_ID from DMS_META_INSTANCE i JOIN
cluster c on i.CLUSTER_ID = c.CLUSTER_ID where INST_TYPE = 'CN'),
collection as (SELECT c.VIRTUAL_CLUSTER_ID, c.CLUSTER_ID,
METRIC_TABLE_NAME,INIT_COLLECTION_TIME from DMS_META_COLLECTION_CONFIG cc JOIN
cluster c on cc.CLUSTER_ID = c.CLUSTER_ID where cc.PLUGIN_TYPE = 'metric' and
cc.COLLECTION_TYPE = 'CN,ALL')
SELECT INIT_COLLECTION_TIME * 1000, c.VIRTUAL_CLUSTER_ID, c.CLUSTER_ID,
METRIC_TABLE_NAME, INST_NAME, HOST_ID, INST_ID, '-1' from inst i JOIN collection c on
i.CLUSTER_ID = c.CLUSTER_ID;
```

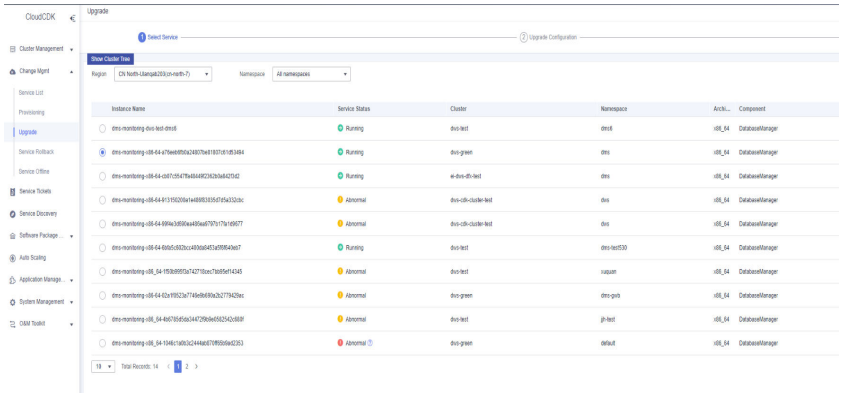
Step 4 Switching the stream database

1. Prepare the password ciphertext for connecting to the GaussDB(DWS) stream data warehouse.
   a. Log in to the **dws-maintain** container.
      ```
      kubectl exec -it `kubectl get pods --all-namespaces | grep maintain | grep Running | head -1 |
      awk '{print $2}'` bash -n `kubectl get pods --all-namespaces | grep maintain | grep Running |
      head -1 | awk '{print $1}'`
      ```
   b. Generate the SCC ciphertext of the password for user **dbadmin** of the GaussDB(DWS) stream data warehouse.
      ```
      cd opsTool
      java -jar SccTool.jar
      1 ${dws_iot_pwd}
      ```
   c. Save the encrypted string.
2. Prepare CDK parameters.
   a. Prepare dms-collection parameters.
      ```
      "spring.datasource.dws.username": "dbadmin",
      "spring.datasource.dws.password": "${scc_encrypt_passwd}",
      "spring.datasource.dws.url": "jdbc:gaussdb://${elb_ip}:${port}/dms?connectionExtraInfo=false"
      "spring.datasource.dws.driverClassName": "com.huawei.gauss200.jdbc.Driver",
      "dms.datasource.primary": "dws",
      "LIVENESS_PROBE.PERRIOD_SECONDS": "66"
      ```
   b. Prepare dms-monitoring parameters.
      ```
      "spring.datasource.dws.username": "dbadmin",
      "spring.datasource.dws.password": "${scc_encrypt_passwd}",
      "spring.datasource.dws.url": "jdbc:gaussdb://${elb_ip}:${port}/dms?connectionExtraInfo=false"
      "spring.datasource.dws.driverClassName": "com.huawei.gauss200.jdbc.Driver",
      "dms.datasource.primary": "dws",
      "LIVENESS_PROBE.PERRIOD_SECONDS": "66",
      "enable.ts.mapper": "true"
      ```

      NOTE

      - **${scc_encrypt_passwd}** indicates the password ciphertext generated in **4.1**.
      - **${elb_ip}** indicates the IP address of ELB.
      - **${port}** indicates the port number of the stream data warehouse.
3. Upgrade microservices.
   a. Upgrade the dms-collection microservice.
      i. Log in to CloudScope using the **op_cdk_sso** account.

ii. Choose **Services** > **CloudCDK** > **Change Mgmt** > **Upgrade**, select **dms-collection**, and click **Next**.



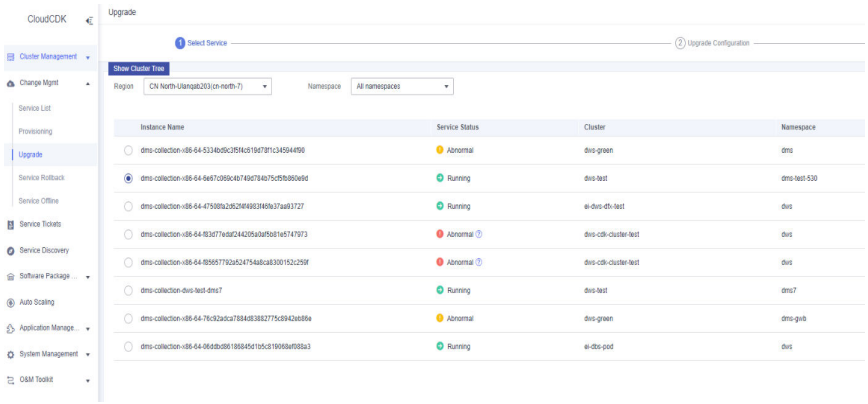
iii. Modify the parameter values based on the CloudAutoDeploy-CDK parameters prepared in **4.2**.

iv. Click **Next** and submit the upgrade on the information confirmation page.

b. Upgrade the dms-monitoring microservice.

i. Log in to CloudScope using the **op_cdk_sso** account.

ii. Choose **Services** > **CloudCDK** > **Change Mgmt** > **Upgrade**, select **dms-collection**, and click **Next**.



iii. Modify the parameter values based on the CloudAutoDeploy-CDK parameters prepared in **4.2**.

iv. Click **Next** and submit the upgrade on the information confirmation page.

Step 5 Verify the switchover.

1. Verify the function of monitoring existing clusters.

Table 2-1 Monitoring items of existing clusters

Item	Expected Result
Cluster Overview	On the **Monitoring Panel** page, the real-time metric curves are properly displayed.

Item	Expected Result
Host Monitoring	Host monitoring: Data on the relative three pages is normal.
Monitoring Settings	Monitoring settings: The monitoring information is displayed properly, and collection items can be disabled or enabled.
Performance Monitoring	Performance monitoring: The curve monitoring panels are not lost and can be added or deleted.
DDL Audit	DDL audit: The DDL audit columns are normal. Audit requests can be made and the request results can be obtained.

2. Verify the function of monitoring new clusters.

Table 2-2 Monitoring items of new clusters

Item	Expected Result
Cluster Overview	On the **Monitoring Panel** page, the real-time metric curves are properly displayed.
Host Monitoring	Host monitoring: Data on the relative three pages is normal.
Monitoring Settings	Monitoring settings: The monitoring information is displayed properly, and collection items can be disabled or enabled.
Performance Monitoring	Performance monitoring: The curve monitoring panels are not lost and can be added or deleted.
DDL Audit	DDL audit: The DDL audit columns are normal. Audit requests can be made and the request results can be obtained.

----End

3 Visualized Dashboard Analysis (QueryBoost)

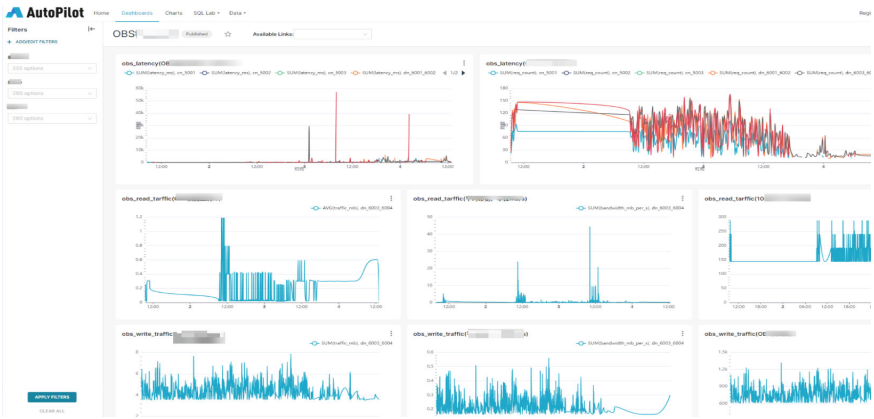
3.1 Overview

QueryBoost is a zero-code report generation tool. It can generate reports on GaussDB(DWS) data through drag and drop.

The process of creating a report is as follows:

1. Establish a connection to the GaussDB(DWS) database.
2. Select a table or view in the database as the data source.
3. Drag the columns of the data source to the corresponding positions in the chart.
4. Drag multiple charts to the same canvas to form a dashboard.

Dashboard examples:



3.2 Preparations

Before using QueryBoost, create a GaussDB(DWS) cluster, obtain an O&M account of the CloudScope platform, and add the account to the corresponding user group.

Creating a GaussDB(DWS) Cluster

Step 1 Log in to the GaussDB(DWS) console and click **Create Cluster**.

Step 2 On the **Create Cluster** page, select a region, product type, and complete the configurations.

Step 3 On the **Create Cluster** page, set **Cluster Name**, **Administrator Account**, **Administrator Password**, and **Database Port**, and click **Create Now**.

----End

Logging in to QueryBoost

Access the QueryBoost access address. The CloudScope login page is displayed. Enter the username and password to log in. The QueryBoost automatically grants permissions to the user based on the group to which the user belongs.

Method 1: Logging In to CloudScope

Step 1 Log in to CloudScope using a browser as a system administrator.

- URL: **https://***Address_for_accessing_CloudScope*, for example, **https://cloudscope.demo.com**
- For details about the URL for accessing CloudScope, see the COP information on the "Portal" sheet of the deployment parameter table exported from HCC Turnkey during Auto Change Platform installation.
- Default account: **op_cdk_sso**
- To obtain the default password of the account, search for the default password of the account on the "CloudScopeLite" sheet of the ***Huawei Cloud Stack 8.5.0 Account List***.

Step 2 In the navigation tree on the top of the page, choose **CloudAutoDeploy-CDK**. The CloudAutoDeploy-CDK page is displayed.

Step 3 In the navigation pane on the left, choose **Change Mgmt** > **Upgrade**.

Step 4 Search for **CONSOLE_ENDPOINT** and record its address.

Step 5 Log in to QueryBoost at **https://{Endpoint}//superset/welcome/**.

----End

Method 2: Logging In to ManageOne Maintenance Portal ServiceCM.

NOTE

To access the QueryBoost page on Huawei Cloud Stack 8.3.1 or later, log in to ServiceCM. However, for versions prior to 8.3.1, you will need to log in to Service OM instead.

Step 1 Log in to ManageOne Maintenance Portal via **https://***ManageOne Maintenance Portal URL*:**31943**. Alternatively, log in to the unified portal and choose **OperationCenter**.

- Password login: Enter the username and password of the account.
  - Default account: **bss_admin**

NOTE

For ManageOne upgraded from 8.2.0 or earlier, the default username is **admin**. For ManageOne 8.2.1 or later, the default username is **bss_admin**.

- Preset password: See the preset password of the ManageOne Maintenance Portal account on the "Type A (Portal)" sheet in ***Huawei Cloud Stack 8.5.0 Account List***.

- Login using a USB key: Insert a USB key with preset user certificates, select the required device and certificate, and enter the PIN.

Log in to ManageOne Maintenance Portal.

Step 2 In the **Common Links** area, click **ServiceCM**. Select your region to access the ServiceCM page.

Step 3 Jump to QueryBoost:

1. In the navigation pane, choose **Monitoring Alarms** > **Cluster Monitoring**.
2. Click **Go to QueryBoost**. The QueryBoost page is displayed.

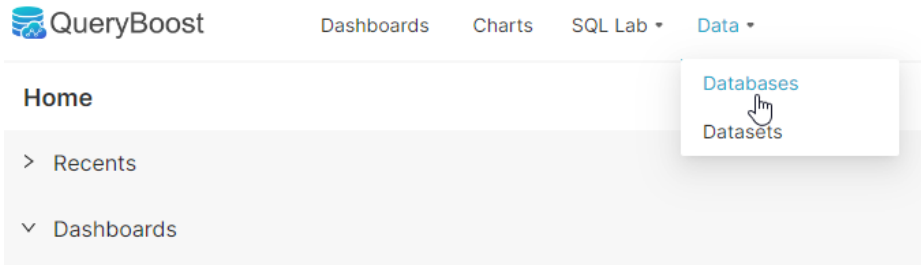
----End

3.3 Obtaining Data

3.3.1 Data Source Connection

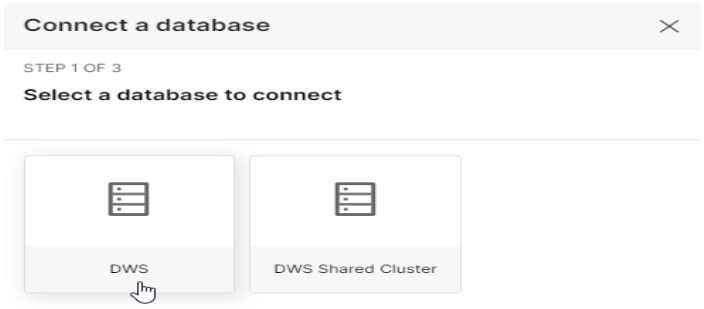
Before report development, you need to add a GaussDB(DWS) connection as the data source for subsequent operations. The procedure is as follows:

Step 1 On the top menu bar, click **Data** > **Databases**. The data source list page is displayed.



Step 2 On the data source list page, click 

in the upper right corner to add a data source. Click **DWS**.



Step 3 On the information configuration page, enter the data source connection information as prompted and click **Connect**.

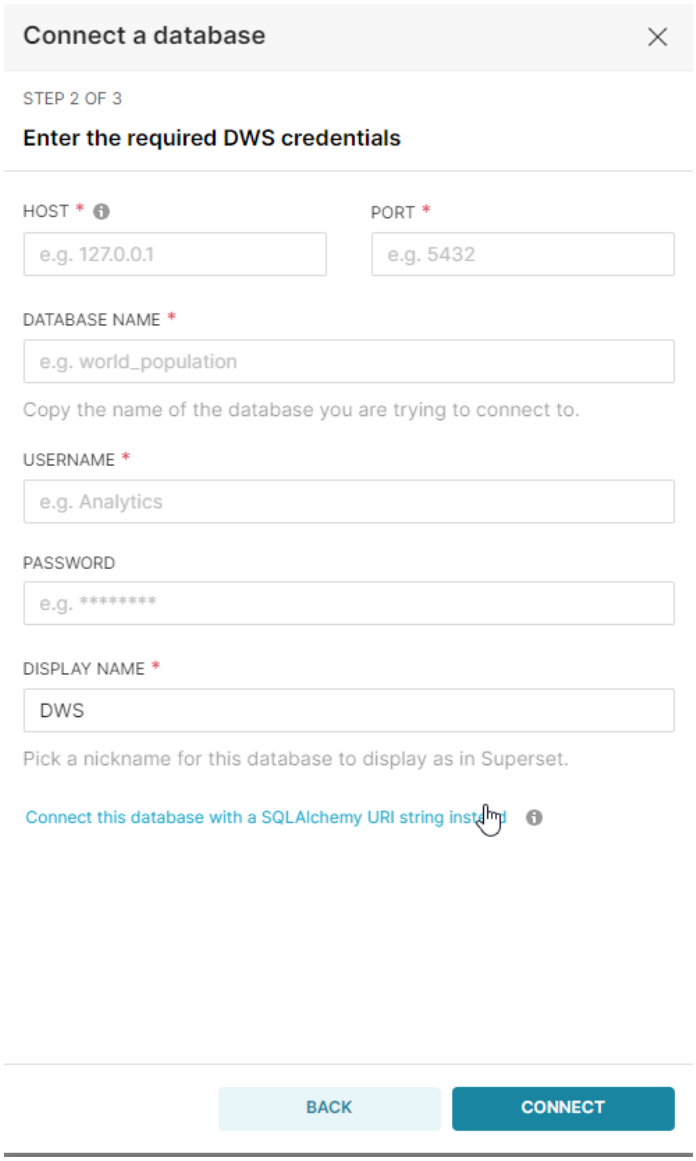
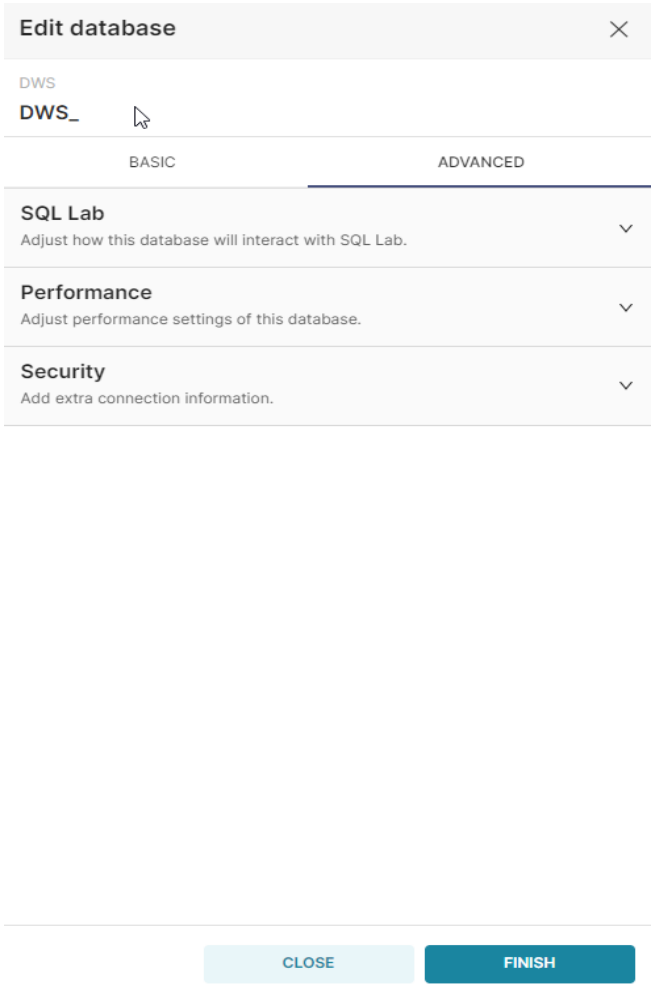


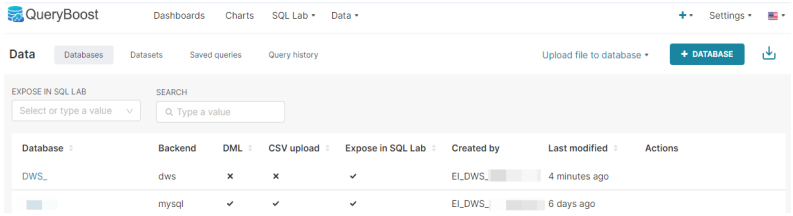
Table 3-1 Connection details

Parameter	Description	Example Value
Host	Public network address of GaussDB(DWS). It can be an IP address or a domain name. If no GaussDB(DWS) cluster has been created, create a cluster on the GaussDB(DWS) console.	127.0.0.1
Port	Public network port of GaussDB(DWS).	8000
Database Name	Name of the database. If the database does not exist in the current cluster, the connection fails. Create a database in the current cluster.	testdb
Username	Username of the GaussDB(DWS) cluster.	-
Password	Password of the GaussDB(DWS) cluster.	-
Display Name	Name displayed on the database list page and in subsequent operations.	dws-bi

Step 4 After the connection is successful, the following configuration page is displayed. Click **Finish** to add the data source.



Step 5 After the data source is successfully configured, you can view the configured data source on the database list page, including the type, creation time, and modification time.



NOTE

The default connection settings are exposed in **SQL Lab** (available in the SQL editor). **DML** and **CSV** files cannot be uploaded. To change the settings, go to **Advanced Settings**.

Table 3-2 Parameter description

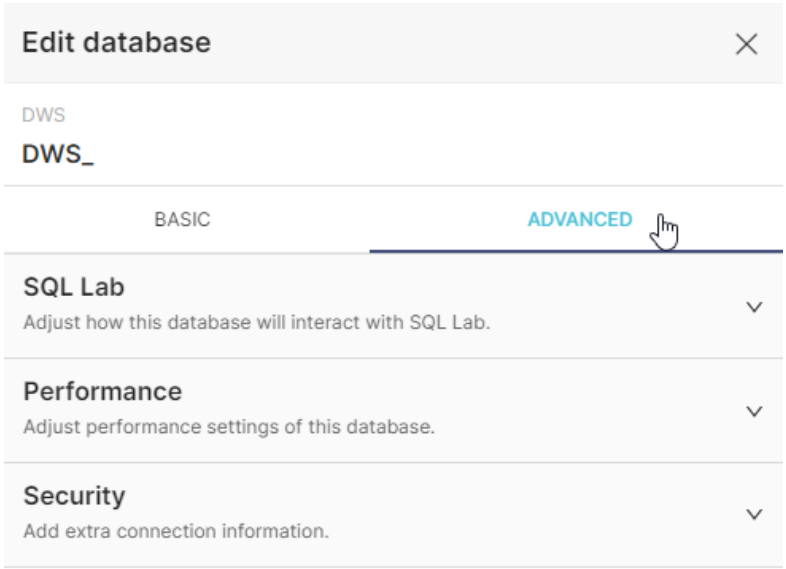
Parameter	Description	Example Value
Databases	Display name of the connected database.	demo

Parameter	Description	Example Value
Backend	DB engine.	dws
DML	Whether DML statements can be executed on the database.	-
CSV Upload	Whether CSV files can be uploaded to the database.	-
Expose in SQL Lab	Whether to expose the database in SQL Lab. You can operate the database in the SQL Editor.	-
Created By	Creator of the database connection.	Admin
Last Modified	Last time the database connection was modified.	{2} days ago
Actions	Operations such as deletion and editing.	-

----End

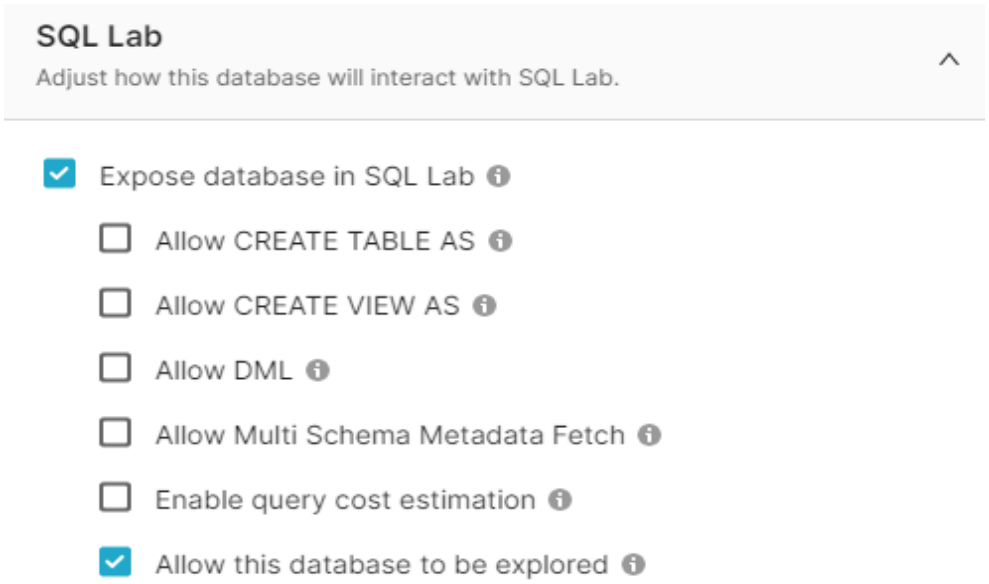
3.3.2 Advanced Settings

To further configure the data source, click  next to the data source in the data source list. On the displayed page, click **Advanced**. The **SQL Lab** and **Performance** configuration items are displayed.



SQL Lab

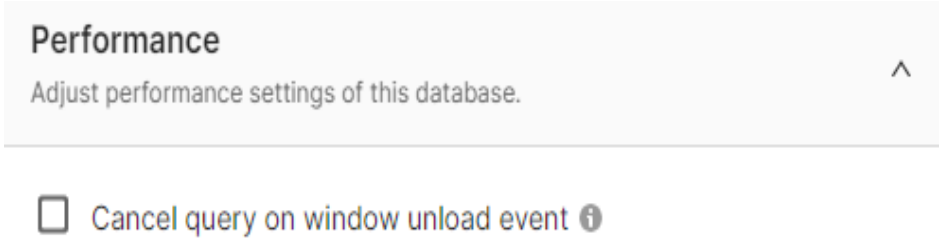
First, click **SQL Lab** to expand the panel. These options are used to configure how your database interacts with the SQL Lab. The options are as follows:



- **Expose Database in SQL Lab**: allows you to use the database in the SQL Lab. This option is enabled by default.
- **Allow CREATE TABLE AS**: allows you to create new tables based on SQL queries.
- **Allow CREATE VIEW AS**: allows you to create new views based on SQL queries.
- **Allow DML**: allows you to use Non-SELECT statements (such as **UPDATE**, **DELETE**, and **CREATE**) to operate the database.

- **Allow Multi Schema Metadata Fetch**: allows the SQL editor to obtain tables and views in all schemas of the database. For a large data warehouse with thousands of tables, this can put burdens on performance as well as the system.
- **Enable query cost estimation**: displays a button that enables you to calculate the cost of a query before running it.
- **Allow this database to be explored**: allows you to see SQL editor results in **Chart Exploration**.

Performance Panel



- **Cancel query on window upload event**: When the browser window is closed or navigated to another page, the running query is terminated. It can be used for GaussDB(DWS) databases.

3.3.3 Troubleshooting Connection Faults

The following table lists the error codes that may occur during data connection or access.

Table 3-3 Error code reference

Error Code	Message	Description
1000	The data source is too large.	Your data source is too large, and the current query and is timeouting. You can solve this problem by reducing the size of the data source or modifying the query to process subsets of the data.
1001	The database is under abnormal load.	Your query may have timed out due to an unusually high load on the database engine. You can make the query simpler, or wait until the database is less loaded and try again.
1002	The database returns an unexpected error.	Your query failed because of a database error, which could be caused by a syntax error, a problem with the query, or an internal issue with the database. Typically, this is not caused by QueryBoost, but rather by the database that is handling your query.

Error Code	Message	Description
1003	Syntax errors exist in the SQL query. Maybe there are spelling mistakes or typos.	Your query failed due to a syntax error in the query. Verify that all columns or tables referenced in the query exist and are spelled correctly.
1004	The column is deleted or renamed in the database.	Your query fails because it references a column that no longer exists in the underlying data source. You should modify the query to refer to another column, or remove this column from the query.
1005	The table is deleted or renamed in the database.	Your query fails because it references a table that no longer exists in the underlying database. You should modify the query to refer to the correct table.
1006	One or more parameters specified in the query are lost.	Your query was not submitted to the database because one or more parameters are missing. You should define all the parameters referenced in the query in a valid JSON document. Check whether the parameter spelling is correct and whether the syntax of the document is valid.
1007	The provided host name cannot be resolved.	The host name provided when a new database is added is invalid and cannot be resolved. Check whether the host name contains spelling errors.
1008	The port is disabled.	The port provided when a new database is added is not enabled. Check whether the port number is correct and whether the database is running and listening on the port.
1009	The host may have been shut down and cannot be accessed on the provided port.	The host provided when the new database is added is not started. It cannot be accessed through the provided port. Check whether there are firewall rules that block access to the host.
1010	QueryBoost encountered an error while running the command.	QueryBoost encountered an error while running the command. Contact the administrator.
1011	The QueryBoost encounters an unexpected error.	Something unexpected have happened to the QueryBoost backend. Contact the administrator.

Error Code	Message	Description
1012	The username provided for database connection is invalid.	The username does not exist in the database. Check whether the username is correct and exists in the database.
1013	The password provided for database connection is invalid.	The password provided by the user is incorrect. Check whether the password is correct.
1014	The username or password is incorrect.	The username does not exist or the password is incorrect. Check whether the username and password are correct.
1015	The database is misspelled or does not exist.	The database is incorrectly written or does not exist. Check whether the input is correct.
1016	The schema is deleted or renamed in the database.	The schema has been deleted or renamed. Check whether the schema is correct and exists.
1017	The user does not have the permission to connect to the database.	The user cannot connect to the database. Ensure that your service account has the viewer and job user roles in the database.
1018	One or more database configuration parameters are missing.	Some parameters required for testing, creating, or editing a database do not exist. Check the required parameters.
1019	The format of the submitted payload is incorrect.	Check whether the format of the request payload is correct (for example, whether it is in JSON format).
1020	The format of the submitted parameter is incorrect.	Check whether the parameter format in the request is correct.
1021	The result backend required for asynchronous query is not configured.	Your QueryBoost instance is not configured with the result backend required for asynchronous queries. Contact the administrator.

Error Code	Message	Description
1022	Data operations are not allowed in the database.	The database supports only SELECT statements. If you need to run Data Manipulation Language (DML) statements on this database, contact the administrator.
1023	The CREATE TABLE AS SELECT (CTAS) statement does not end with a **SELECT** statement.	The last statement of a CTAS query must be a **SELECT** statement. Ensure that the last statement in the query is a **SELECT** statement.
1024	The CVAS (create view as select) query has multiple statements.	The CREATE VIEW AS SELECT (CVAS) query should have only one statement. Ensure that the query contains only one statement and there is no other semicolons except that at the end of the last statement.
1025	The CVAS query is not a **SELECT** statement.	The CVAS query should be a **SELECT** statement. Ensure that the query contains only one **SELECT** statement.
1026	The query is too complex and takes a long time.	The submitted query is complex and cannot run within the time limit defined by the QueryBoost administrator. Check your query and optimized it.
1027	Too many queries are running in the database.	The database may be overloaded with too many queries. Try again later or contact the administrator.
1028	The format of one or more parameters specified in the query is incorrect.	The query contains one or more template parameters that are in incorrect formats. Check your query and make sure that all template parameters are enclosed in double parentheses, for example, {{ ds}}. Then, execute your query again.
1029	The object does not exist in the database.	A schema, column, or table does not exist in the database.
1030	The query may have syntax errors.	The query may have syntax errors. Check the query syntax and run again.
1031	The result backend no longer has data from the query.	The query result may be deleted from the result backend after a period of time. Run your query again.
1032	The query associated with the result has been deleted.	The query associated with the stored result no longer exists. Run your query again.

Error Code	Message	Description
1033	The results stored in the backend are stored in different formats and cannot be deserialized.	Query results are stored in a format that is no longer supported. Run your query again.
1034	The provided database port is invalid.	Ensure that the provided database port number is an integer ranging from 0 to 65535.
1035	The remote worker operations cannot be queried.	The query is not started by an asynchronous worker. Contact your administrator for assistance.
1036	The database is deleted.	Operation failed because the referenced database no longer exists. Contact your administrator for assistance.

3.4 Preparing Data

3.4.1 Overview

Creating data sets is a prerequisite for report development. All data displayed in charts comes from data sets, which are created from data in the database.

QueryBoost offers various methods to create datasets, which can be saved and used for future report development.

- **Creating a Physical Dataset**
- **Creating a Virtual Dataset**

To create a dataset, you must first connect to the data source. If you have not done so already, refer to **Data Source Connection**. Once connected, follow the steps provided in this section to create your dataset.

NOTE

- QueryBoost does not extract data from your database.
- All storage and computing takes place at your backend.
- QueryBoost is just a connection layer on top of the back end. A dataset is an abstraction of data in a database.
- All charts are based on datasets.

3.4.2 Physical Datasets

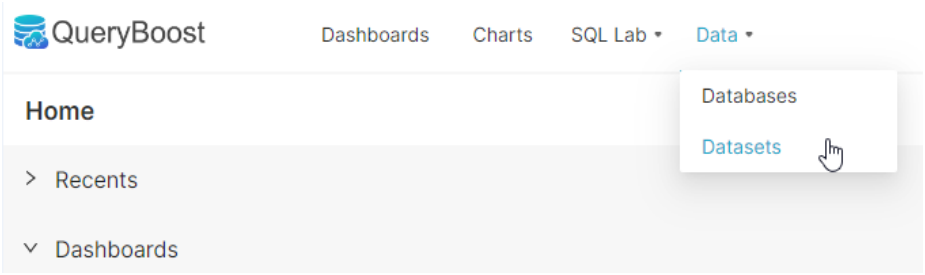
A physical dataset corresponds to a table in a connected database.

In QueryBoost, you can quickly add a physical dataset and access it. The procedure is as follows:

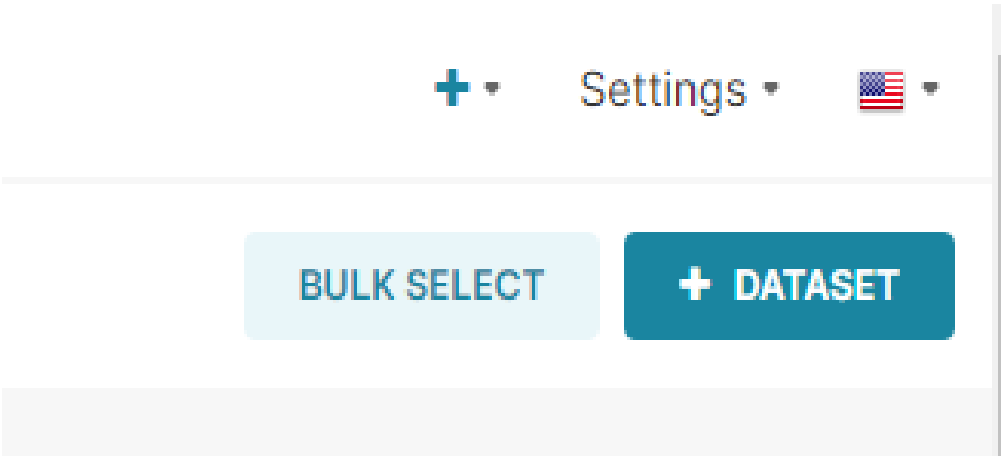
- **Adding a Physical Dataset**
- **Accessing a Physical Dataset**

Adding a Physical Dataset

Step 1 On the toolbar, choose **Data** > **Datasets**.

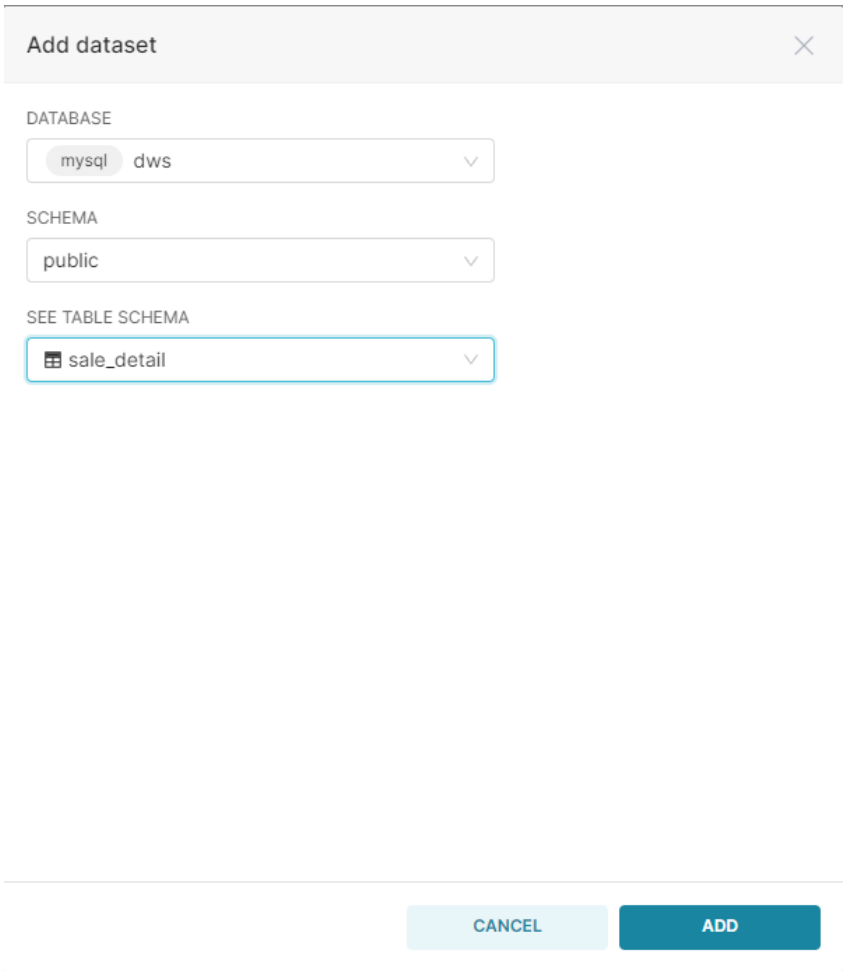


Step 2 Then, select **+ DATASET**.



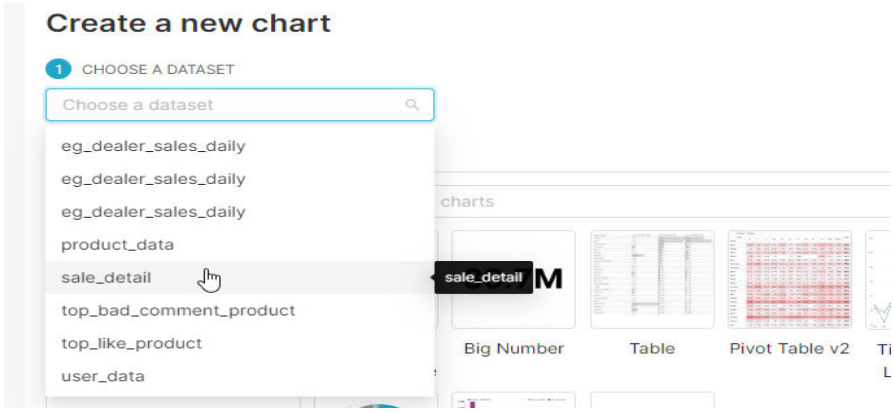
Step 3 The **Add dataset** dialog box is displayed. In the following figure, the **sale_detail** dataset is added.

- In the **DATABASE** field, select a database. After the database is selected, the **SCHEMA** field populates all available options.
- In the **SCHEMA** field, select a schema. After the schema is selected, the **SEE TABLE SCHEMA** field populates all available options.
- In the **SEE TABLE SCHEMA** field, select a data table as the dataset.



Step 4 After confirming that the information is correct, click **ADD**. A physical dataset is added.

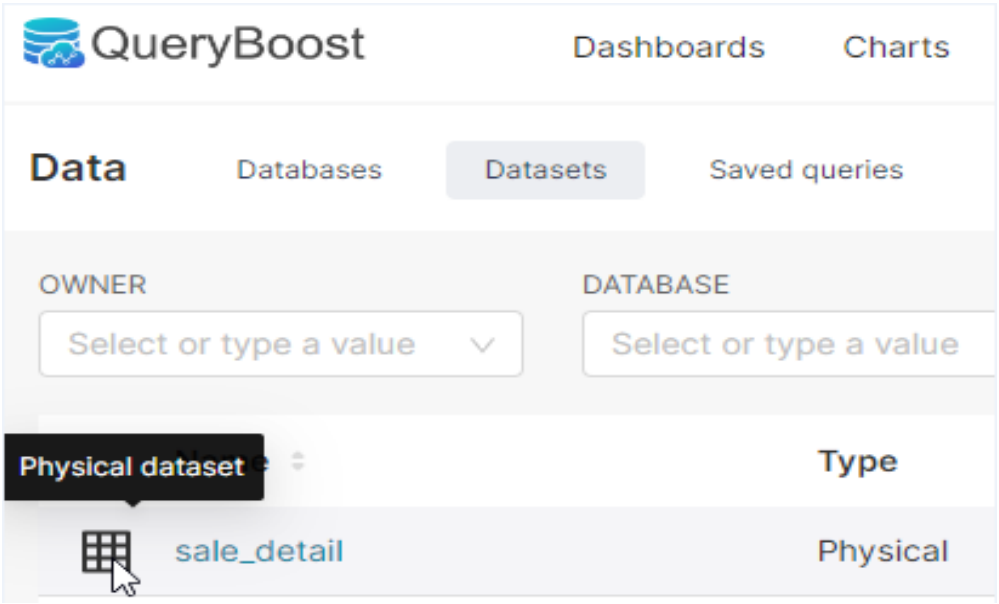
Step 5 When you create a chart, your dataset will be available in the **CHOOSE A DATASET** field. For example, the newly added dataset **sale_detail** will be displayed in the **CHOOSE A DATASET** drop-down list.



----End

Accessing a Physical Dataset

Step 1 The physical dataset appears as a black grid icon on the **Datasets** page and is also displayed as **Physical** in the **Type** column. For example, in the following figure, the **sale_detail** dataset is a physical dataset.



Step 2 Now that you have created a new dataset, you can proceed with **Creating a Chart**. Select the newly created dataset for the new chart.

----End

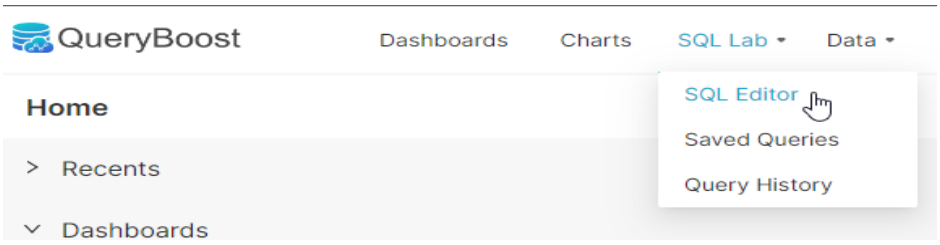
3.4.3 Virtual Datasets

A virtual dataset is a data view created in QueryBoost using the SQL Editor.

Go to the **SQL Editor** page, select a database and schema, write an SQL statement, and execute the query. After the query is executed, click **Explore** on the **Results** page and save the dataset.

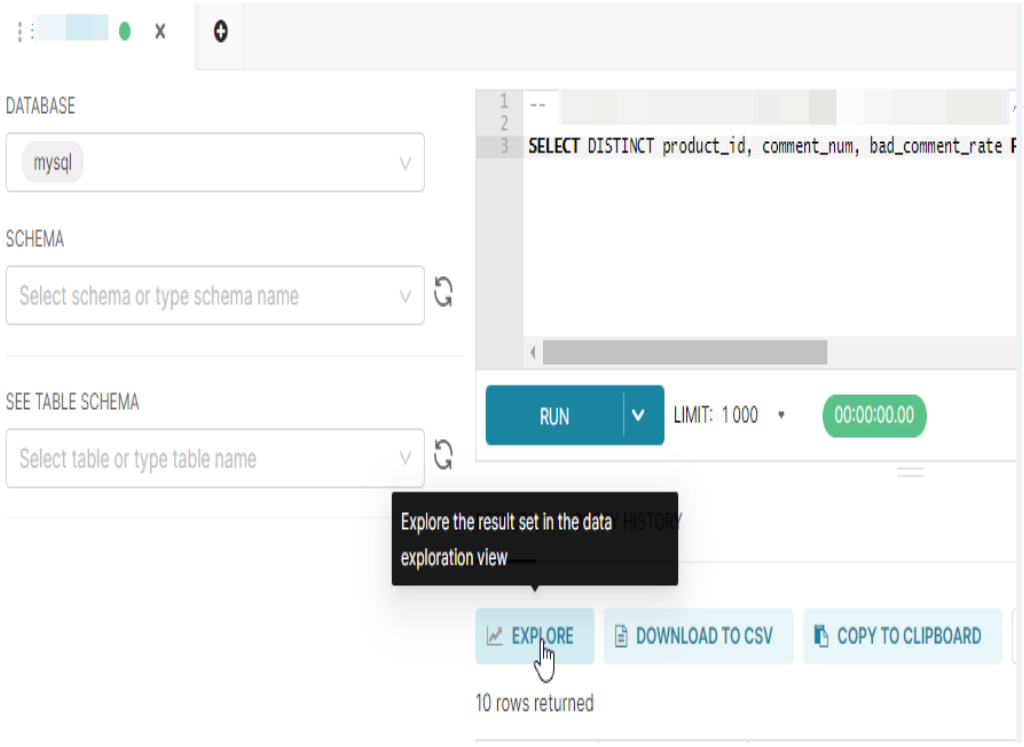
Creating a Virtual Dataset

Step 1 On the toolbar, hover the cursor over **SQL Lab**, and then select **SQL Editor** from the drop-down menu.



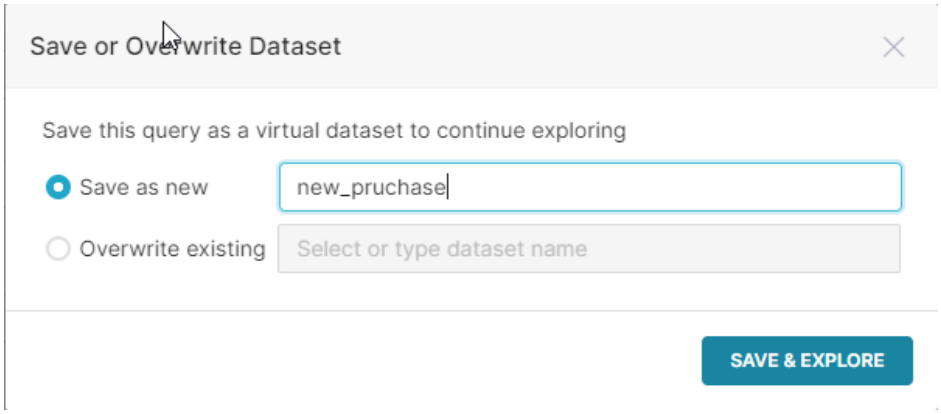
Step 2 Select a database and a schema, enter a query statement in the SQL compilation box, and click **Execute**.

Step 3 On the **Results** tab, click **Explore**, the **Save or Overwrite Dataset** panel is displayed.



Step 4 To save the new virtual dataset, select **Save as new**, and then enter the name of the new virtual dataset in the text box.

Step 5 If you want to overwrite an existing virtual dataset, select **Overwrite existing**, and then enter or select an existing dataset name in the text box.



Step 6 Select **Save & Explore**. The **Chart Exploration** page is displayed. The selected data set is the created virtual data set.

----End

Accessing a Virtual Dataset

Step 1 The virtual dataset appears as a blue grid icon on the **Datasets** page and is also displayed as **Virtual** in the **Type** column. Here, you can manage, explore, delete, and view virtual datasets in the way you manage physical datasets.

Step 2 Now that you have created a new dataset, you can proceed with **Creating a Chart**. Select the newly created dataset for the new chart.

----End

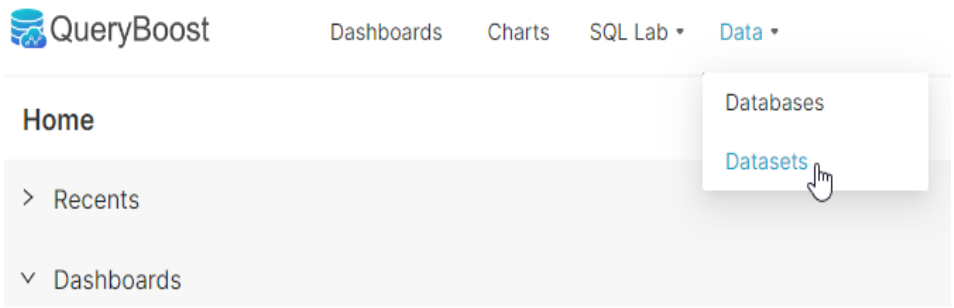
3.4.4 Advanced Settings

3.4.4.1 Setting the Page Position

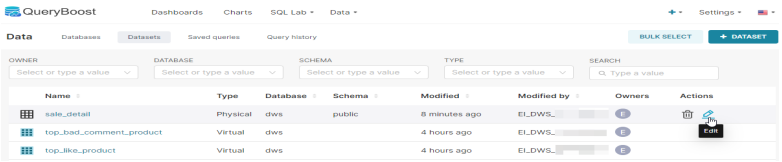
You can further configure the created data sets to meet more report requirements. In each dataset, you can modify its source, metrics, and columns, and create new calculated columns. In QueryBoost, you can access the dataset editing page from two locations to modify datasets: **Datasets** and **Chart Exploration** page.

Datasets page

Step 1 On the toolbar, hover the cursor over **Data**, and then select **Datasets**. The **Datasets** page is displayed.



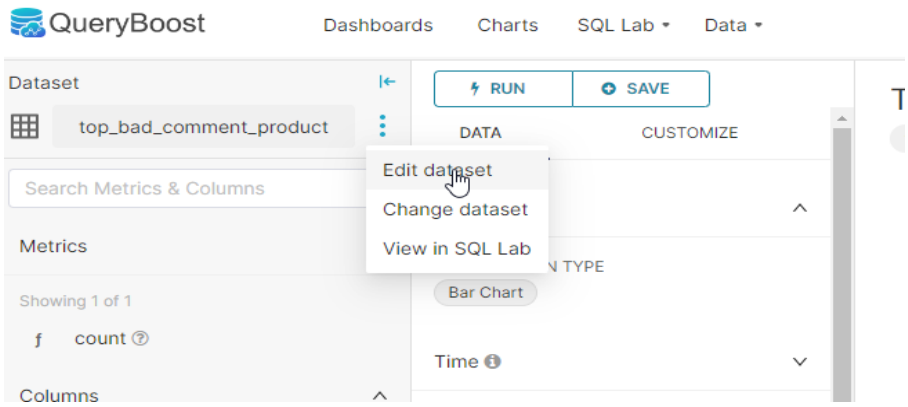
Step 2 This page lists all datasets of the current user. The **Delete** and **Edit** buttons are available in the **Actions** column of each dataset. Click **Edit**. The **Edit dataset** panel is displayed.



----End

Chart Exploration page

When configuring a chart, you can also click the three dots in the **Dataset** panel in the upper left corner and select **Edit dataset**.



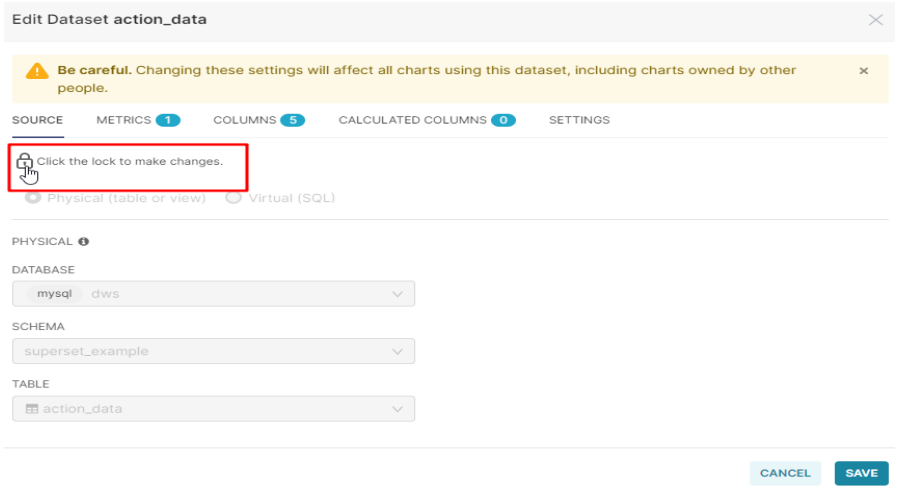
3.4.4.2 Advanced Setting Parameters

3.4.4.2.1 Source

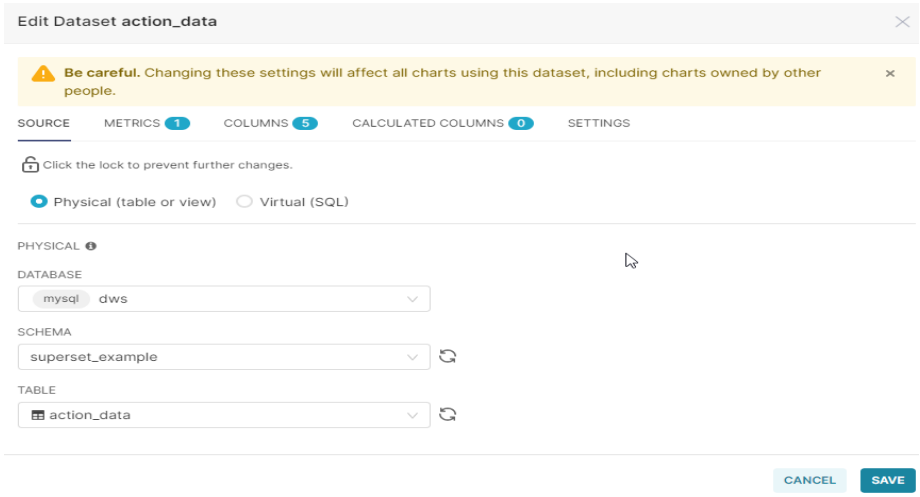
By default, the dataset source cannot be edited. To edit the dataset source, click the lock icon at the top of the page. You are not advised to change the data set source. If you change the data set source, all charts will be affected. If you need a new data set, you can create a new data set.

NOTE

The dataset configuration will affect all charts that use the dataset, including charts of other users. Exercise caution when modifying dataset parameters.



After you click the lock button, the dataset source can be edited.



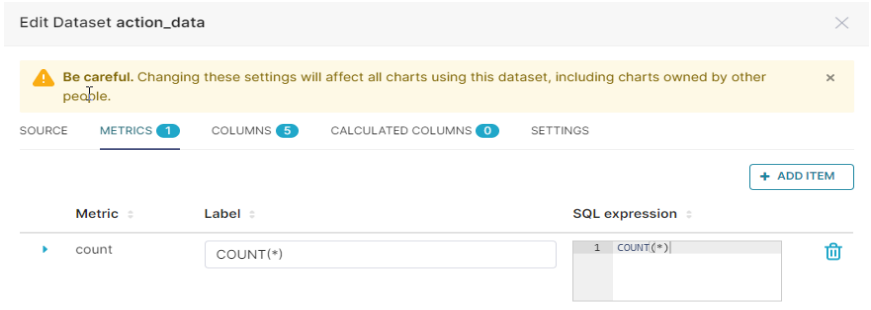
3.4.4.2.2 Metrics

Metrics are used to aggregate grouped data.

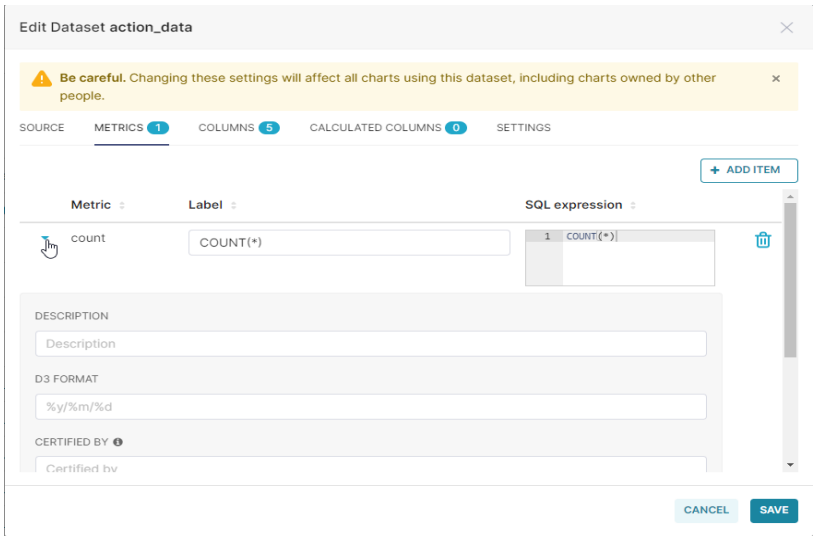
First, in the **Edit dataset** panel, click the **Metrics** tab. All metrics of the current dataset are displayed.

The following fields can be edited in the **Metrics** column:

- **Metric**: indicates the name of a metric.
- **Label**: indicates the label of a metric, which is displayed to users. If no label is set, the metric name is displayed.
- **SQL expression**: SQL expression associated with the metric, for example, **COUNT(*)**.



To edit other fields associated with the metric, select the extension arrow on the left of the metric row.



The editable fields are as follows:

- **Label**: metric label.
- **DESCRIPTION**: A short description of the metric - a message displayed when the user hovers the mouse over the letter **i**.
- **D3 FORMAT**

  D3 format value, you can customize the display format of values and times. For example:

  – The given value is 1234.567.

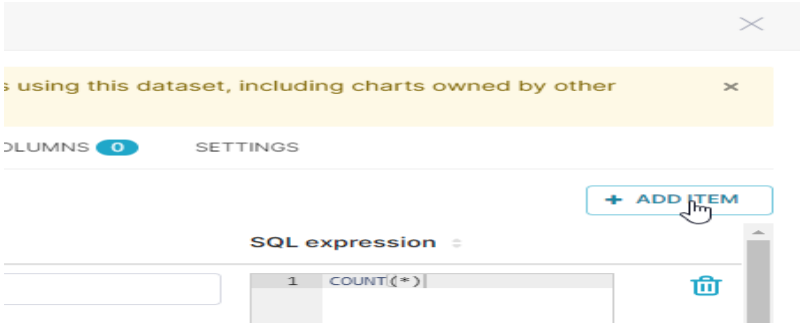
    ▪ .2% = 123456.7%

    ▪ .2k = 1K

    ▪ ,.2r = 1,200
- **CERTIFIED BY**: name of the certification authority/person.
- **Certification Details**: Detailed information about certification.
- **Warning**: A warning message that will appear in the measure selection field after setting.

Adding a Metric

1. To add a new counter, select **+ ADD ITEM** to add a row. A new row is displayed. Enter the metric name and SQL expression to add the metric. To add detailed information, click the extension arrow on the left of the metric line.



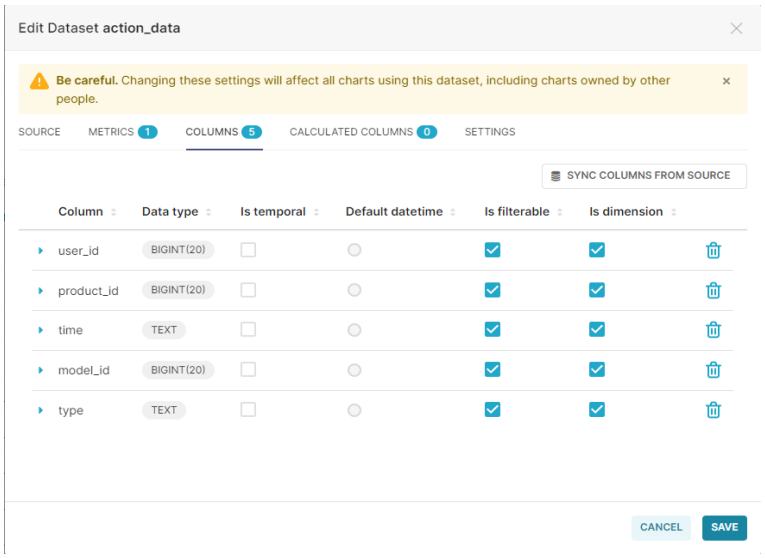
2. Click **Save**.

3.4.4.2.3 Columns

A column is a field in a dataset. The **Columns** tab displays all columns in the dataset.

The column fields are as follows:

- **Column**: column names.
- **Data type**: data type (for example, integer, floating vertex number, variable character, timestamp, etc.).
- **Is temporal** (editable): indicates whether the column is a time column. If this parameter is selected, the column is of the time type and can be used as an option in the **Time** panel.
- **Is filterable** (editable): indicates whether a column can be filtered. If this parameter is selected, the column can be used as an option in the **Filter** panel.
- **Is dimension** (editable): indicates whether the column is a dimension column. If this parameter is selected, the column should be included in the **Groupe** field as an option.
- **Delete**: Click to delete the field.



Select the extension arrow on the left of the column row to define more information about the column.

The editable fields are as follows:

- **Label**: Column label, which serves as the column name displayed to users.
- **DESCRIPTION**: A short description of the column - a message displayed when the user hovers the mouse over the letter **i**.
- **Date and time format:**

  Enter the date and timestamp in Python format. Specifically, the main purposes of calculated columns are:

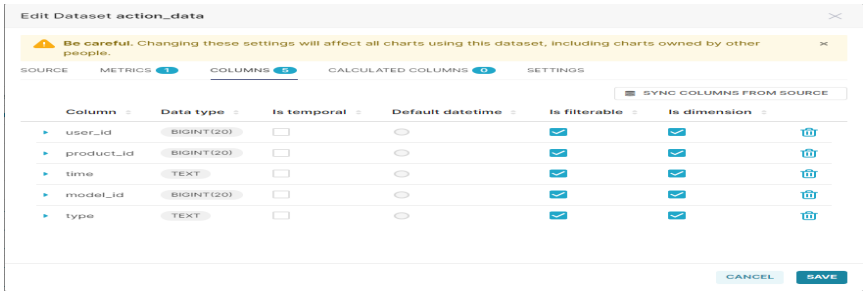
  – %m- %d- %Y: 02-27-2022 (February 27, 2022)

- %a %d, %y: Tue 2, 2022 (Tuesday, February 2022)
- %x %X: 02/27/2022 17:41:00 (17:41:00, February 27, 2022)
- %B- %Y: February-2022 (February 2022)

Table 3-4 Python date format

Symbol	Description
%y	Two-digit year (00-99)
%Y	Four-digit year (000-9999)
%m	Month (01-12)
%d	One day in a month (0-31)
%H	Hours in 24-hour format (0-23)
%I	Hours in 12-hour format (01-12)
%M	Minutes (00-59)
%S	Seconds (00-59)
%a	Localized short day-in-week name
%A	Localized complete day-in-week name
%b	Localized short month name
%B	Localized complete month name
%c	Localized date and time representation
%j	One day in the year (001-366)
%p	Localized A.M. or P.M.
%U	Number of weeks in a year (00-53). Sunday is the start of a week.
%w	Week (0-6). Sunday is the start of a week.
%W	Number of weeks in a year (00-53). Monday is the start of a week.
%x	Localized date representation
%X	Localized time representation
%Z	Name of the current time zone

The **SYN COLUMNS FROM SOURCE** button is used to reconnect to the data source and add new columns and delete columns.



3.4.4.2.4 Calculated Columns

The purpose of calculated columns is to provide more suitable data based on chart requirements. Specifically, the main purposes of calculated columns are:

- **Data conversion**: Change the data to a different format.
- **Data enrichment**: Add meaningful information to the data.
- **Data verification**: Correct and verify data.

The **CALCULATED COLUMNS** tab displays all calculated columns in the dataset.

The column fields are as follows:

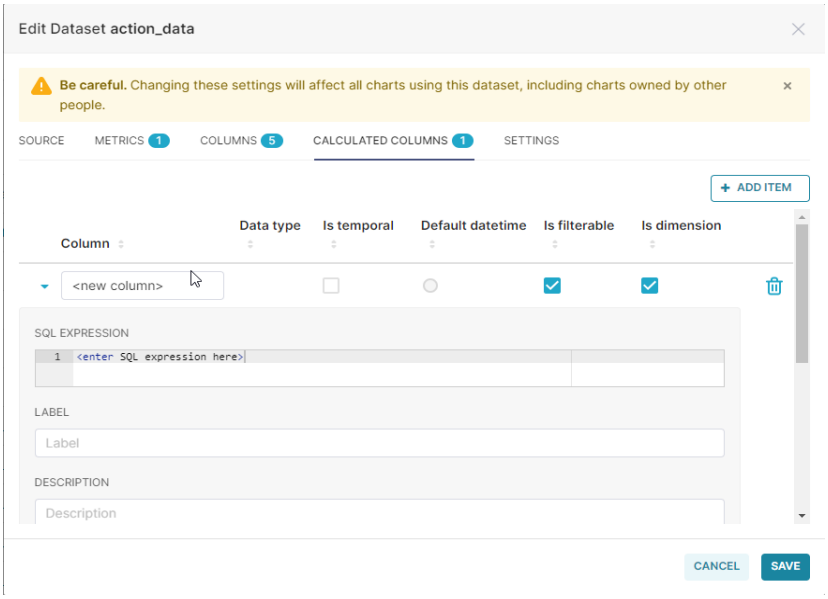
- **Column** (editable): column name.
- **Data type**: data type (for example, integer, floating vertex number, variable character, timestamp, etc.).
- **Is temporal** (editable): indicates whether the column is a time column. If this parameter is selected, the column is of the time type and can be used as an option in the **Time** panel.
- **Is filterable** (editable): indicates whether a column can be filtered. If this parameter is selected, the column can be used as an option in the **Filter** panel.
- **Is dimension** (editable): indicates whether the column is a dimension column. If this parameter is selected, the column should be included in the **Groupe** field as an option.
- **Delete**: Click this button to delete a calculated column.

Click the extension arrow on the left of the calculated column row to define more column information. The editable fields include:

- **SQL expression**: SQL expression associated with the calculated column.
- **Label**: Column label, which serves as the column name displayed to users.
- **DESCRIPTION**: A short description of the column, which is displayed when the user hovers the mouse over the letter **i**.
- **Date and time format:**

  Enter the date and timestamp in Python format. Specifically, the main purposes of calculated columns are:

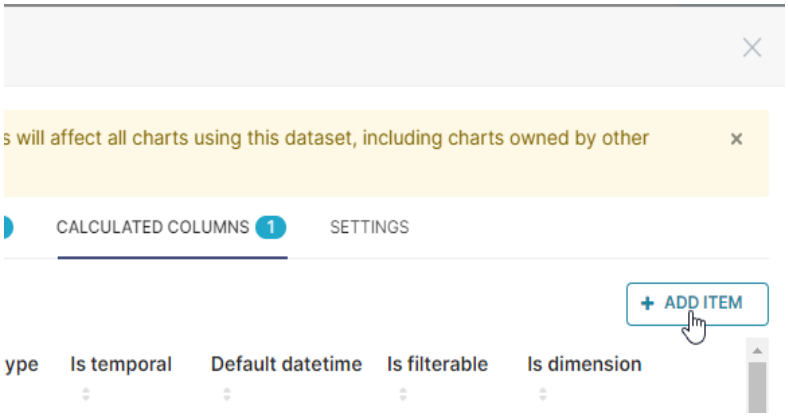
  - %m- %d- %Y: 02-27-2022 (February 27, 2022)
  - %a %d, %y: Tue 2, 2022 (Tuesday, February 2022)
  - %x %X: 02/27/2022 17:41:00 (17:41:00, February 27, 2022)
  - %B- %Y: February-2022 (February 2022)



Adding a Calculated Column

To add a calculated column, select **+ ADD ITEM** to add a row.

A new row will appear. By default, the **Is dimension** and **Is filterable** fields are selected.



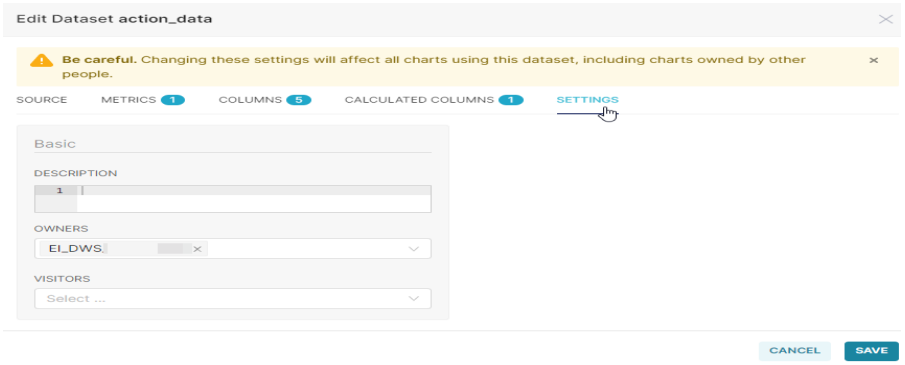
First, under the **Column** field, enter the name of the new calculated column.

Next, in the **SQL Expression** field, enter the SQL expression for the new calculated column.

In the **Data Type** field, select the appropriate data type for the computed column.

3.4.4.2.5 Settings

1. You can set the dataset description and owner.
2. Click **Save**.



3. After the dataset is set, click the dataset name to go to the **Chart Exploration** page and edit the chart.

3.5 Data Analysis

3.5.1 Charts

3.5.1.1 Overview

This chapter describes how to create a new chart and then configure it on the **Chart Exploration** page.

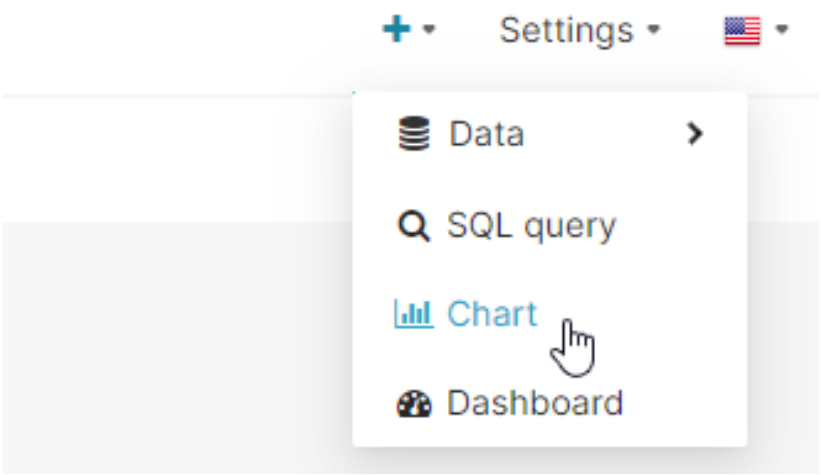
Creating a Chart: Select a data set and a chart type, and click **+ Chart**. The chart exploration page is displayed.

Chart Exploration: Configure charts based on metrics and dimensions, preview the charts, and saves the chart.

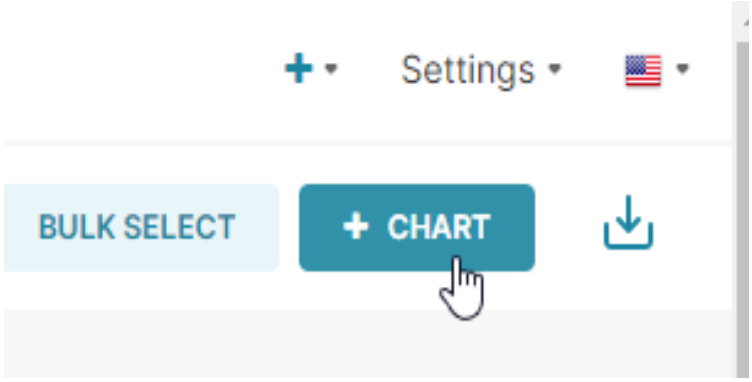
3.5.1.2 Creating a Chart

Creating a Chart

Step 1 On the top toolbar, select the plus sign **+**, and then select **Chart** from the drop-down menu.



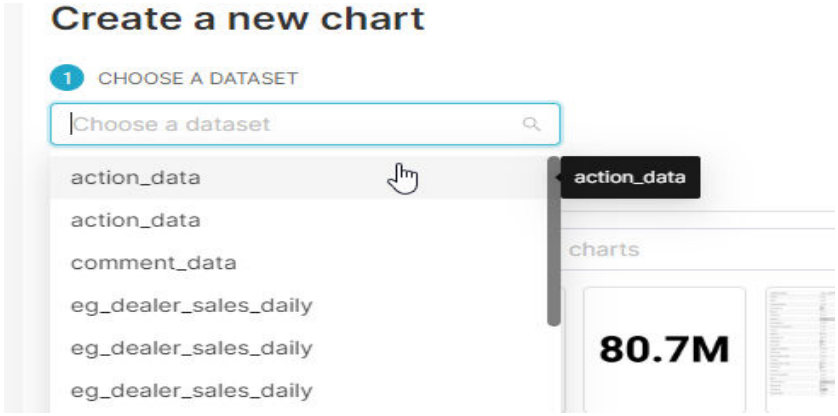
Step 2 You can also go to the chart page on the toolbar and click **+ Chart** to go to the chart creation page.



----End

Selecting a Dataset

Select a dataset from the drop-down list. If no dataset is available, you can go to **Preparing Data** to create datasets.



Selecting a Chart Type

All chart types supported by QueryBoost are displayed on this page. To facilitate search, the page provides the functions of filtering and searching for charts by chart label and category.

The following describes how to filter and search for charts. If you have selected a chart type, skip this section and go to **Chart Exploration** to create a chart.

NOTE

After filter or search, select the **Charts** tab, and then click **Create New Chart** to go to the **Chart Exploration** page.

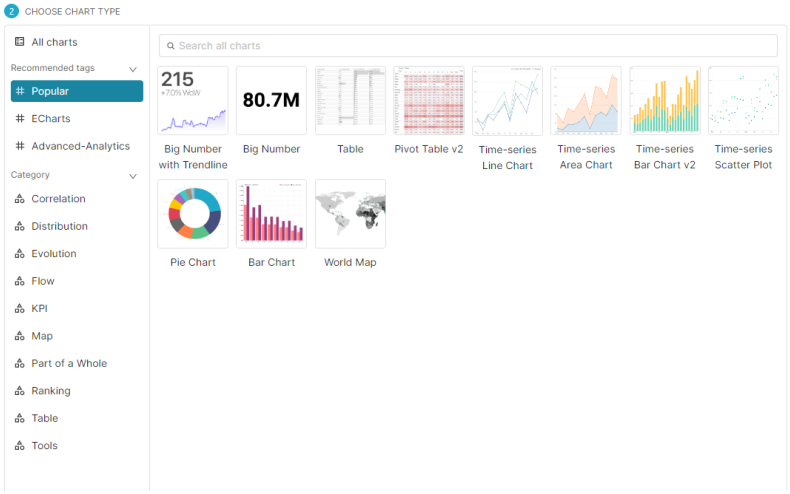
Filtering Charts

The chart filter panel on the left organizes charts based on the following:

- Recommended Tags: By default, charts are filtered based on the **# Popular** tag. Other recommended tags include **# ECharts**, which displays visualized

charts, and **# Advanced-Analytics**, which contains charts with more detailed analysis functions.

- **All charts**: If you select **All charts**, all available charts in QueryBoost are displayed.
- **Category**: You can also filter charts by category.



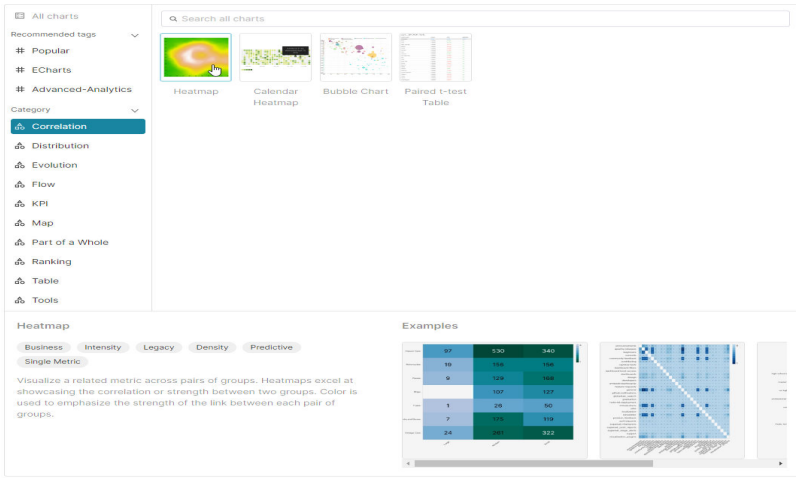
Viewing Chart Information

To learn more about a chart, click the chart card. The details about the chart are displayed in the information panel at the bottom of the page.

Each information panel contains three elements:

- Keywords: Keywords of the chart, which can be used for the search (as described below). In the preceding figure, keywords include **Business**, **Intensity**, **Legacy**, **Density**, **Predictive**, and **Single Metric**.
- Description: description of the chart, including basic description and typical use cases.
- Examples: Thumbnail of the chart.

For example, when you select **Correlation** and then the **Heatmap** chart, its details are displayed at the bottom of the page.



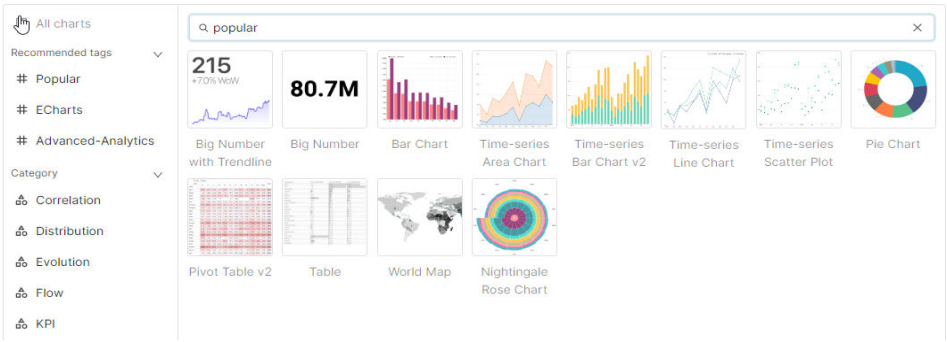
Searching for a Chart

The search function is used to search for charts based on keywords. When you enter keywords in the search box, the results are displayed with each letter you enter.

Searchable text pools include:

- Chart names
- Chart types
- Recommended tags
- Keywords
- Chart description

In the following figure, we select the **new_purchase** dataset and search for **Popular**. Charts that match the keyword are displayed in the results.



3.5.1.3 Chart Exploration

After you create a new chart, the page is redirected to the **Chart Exploration** page, on which you can configure chart metric dimensions and chart styles.

In this section, we will introduce all configuration items and page elements on the chart exploration page in detail, including:

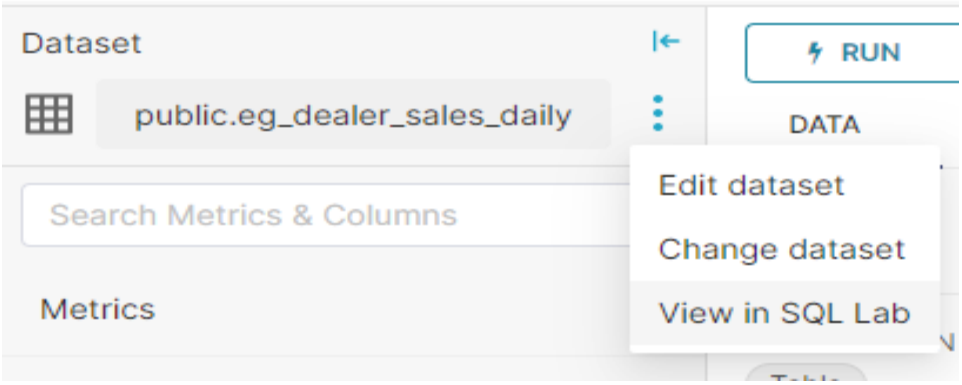
- **Dataset Panel**
- **Metrics and Column Panel**
- **Chart Type Panel**
- **Time Panel**
- **Query Panel**
- **Chart Panel**
- **Data Panel**
- **Customized Configurations**
- **Running and Saving the Query**
- **Exporting Chart Data**

Dataset Panel

The dataset panel displays the dataset selected for the current chart.

Click the vertical ellipsis icon to use more dataset functions.

- **Edit dataset**: modifies metrics, columns, and calculated columns.
- **Edit dataset**: displays all datasets of the current user. You can select new dataset as the dataset of the current chart.
- **View in SQL Lab**: opens the selected dataset in the SQL editor.



Metrics and Column Panel

This panel lists all metrics and columns in the current dataset. If a large data set is used, you can quickly find the required data in the search box.

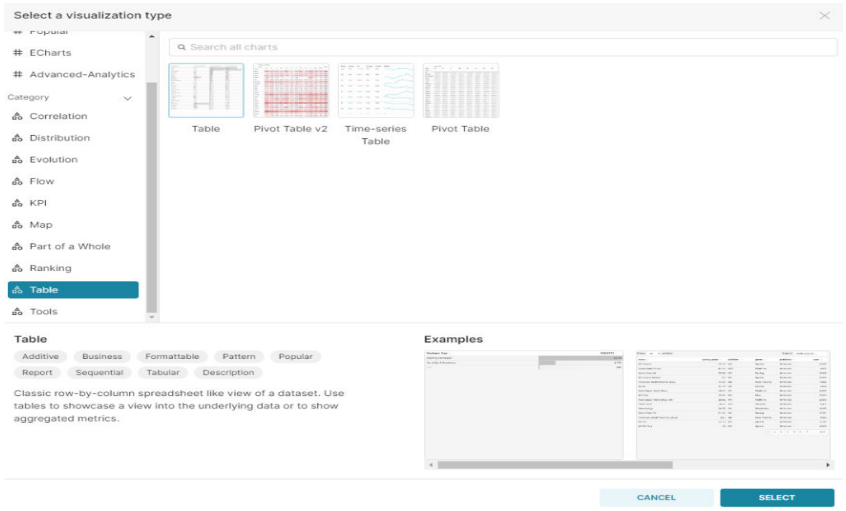
Each data type has a symbol that indicates its type:

- *ƒ*: metric functions
- **Clock icon**: time column of the data source
- **ABC**: text data
- **#**: numeric data

Chart Type Panel

The chart type panel is the same as that on the **Creating a new chart** page. You can change the chart type based on the data display effect when configuring a chart.

To change the chart type, click the current visualization type. The **Visualization Type** panel is displayed, which contains all chart types. The panel functions are the same as those on the page.



Time Panel

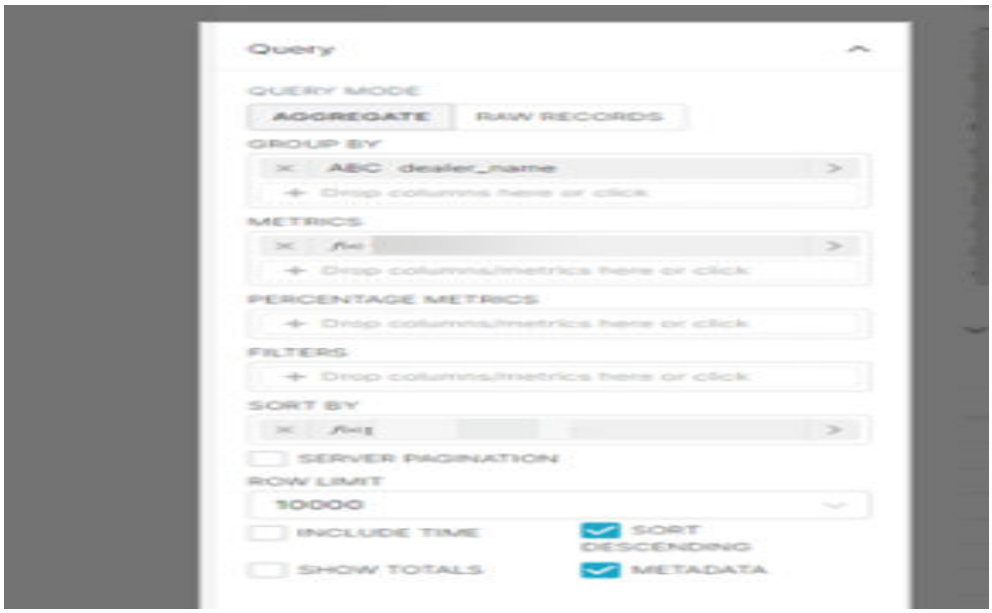
The time panel is used to set time-related elements in a chart.

The time chart attributes are as follows:

- **TIME COLUMN**: indicates the time data column of the chart. The available fields in the time column are the columns of the clock icon in the metrics and column panel.
- **Time Grain**: indicates the time granularity that you want to apply to a chart, for example, day, week, or month.
- **TIME RANGE**: Only the data within the time range specified by this parameter is queried.

Query Panel

The query panel is the main panel for chart queries. The query panel uses different fields for different chart types.



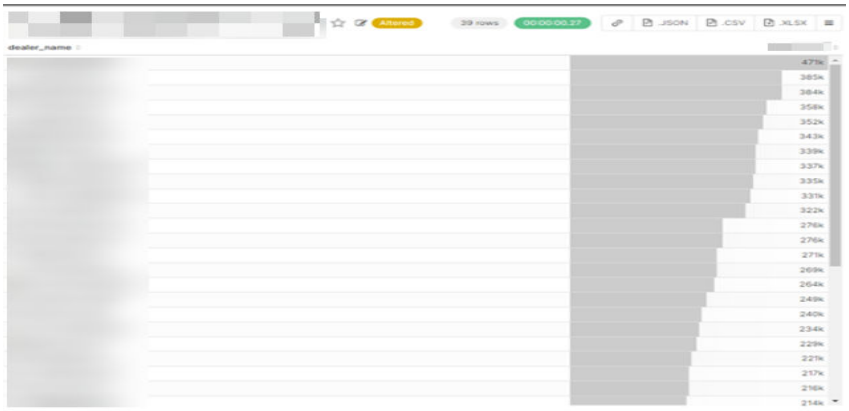
Generally, the query panel includes the following fields:

- **GROUP BY**: indicates the grouping field.
- **Metrics**: measurement values, which are aggregated by group.
- **SORT BY**: used to define the displayed metrics when row limit or sequence limit is applied.
- **FILTERS**: You can add fields as filters.

Chart Panel

The chart panel displays the visualization of data and the following:

- Chart title
- Add-to-favorites
- Chart changes
- Chart data
- Refresh button
- Query processing time

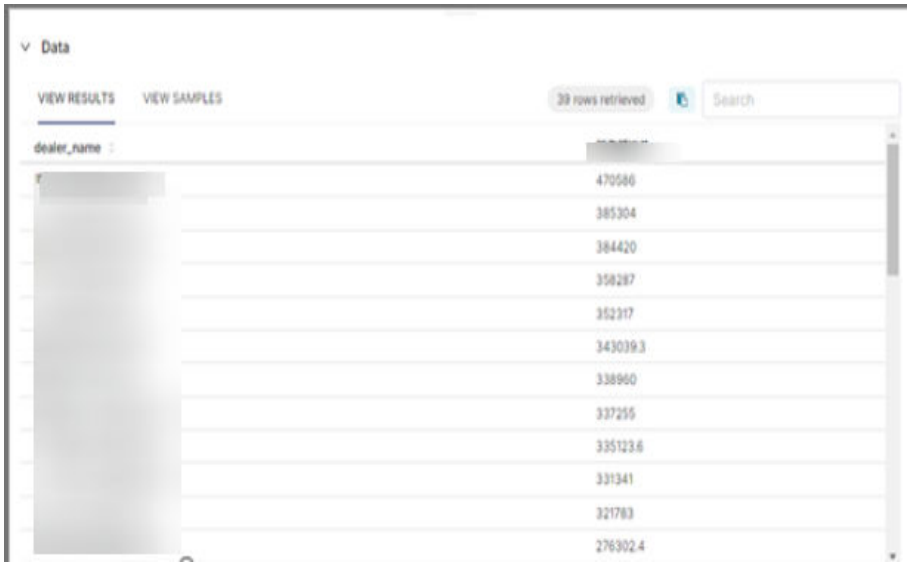


The chart dashboard also displays the following chart tools:

- **Copy chart URL**: Copy the URL of the chart to your clipboard. Ensure that the person you are sharing it with has permission to view the chart.
- **Code embedding**: provides embeddable code of the chart.
- **Export in .JSON format**: exports chart data in JSON format.
- **Export in .CSV format**: exports chart data in CSV format.
- Functions on the drop-down list box:
  - **Edit Attribute**: Modify the chart permission attributes.
  - **Check query**: checks the SQL query to chart data.
  - **Run in SQL editor**: jumps to the **SQL Editor** and load the query statements of the chart in the SQL editor.
  - **Download as image**: Download the chart to the local host in JPEG format.

Data Panel

Below the chart is an extensible section **Data**.

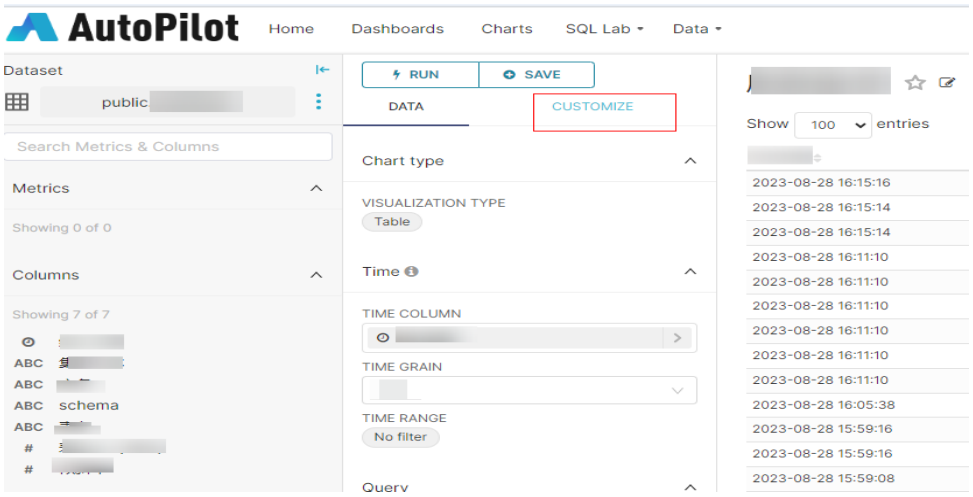


VIEW RESULTS Tab

The **VIEW RESULTS** tab page is displayed by default, on which you can see the query results based on your selected metrics and columns.

Customized Configurations

The **Customized Configuration** page is used to configure the display of charts. Most charts provide customized display options. For example, a table has the following configuration items.



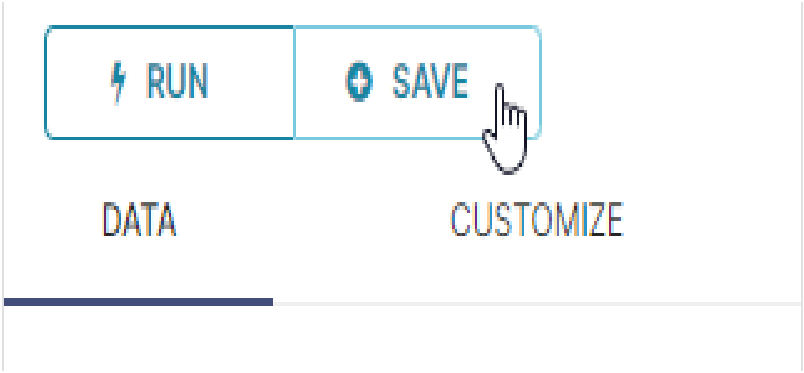
Running and Saving the Query

Running a Query

To run a query, click **Run**. Alternatively, after changing the chart configuration, you can select **Run Query** in the middle of the chart panel.

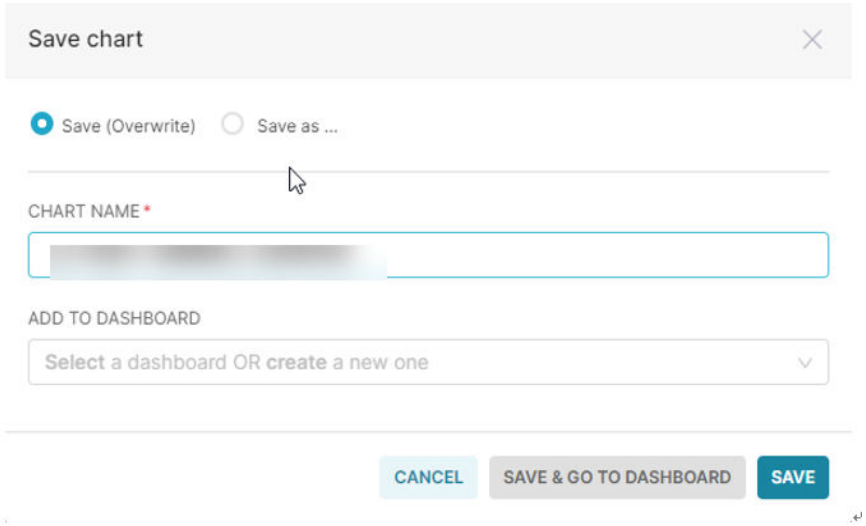
Saving a Chart

To save the chart, click **Save**.

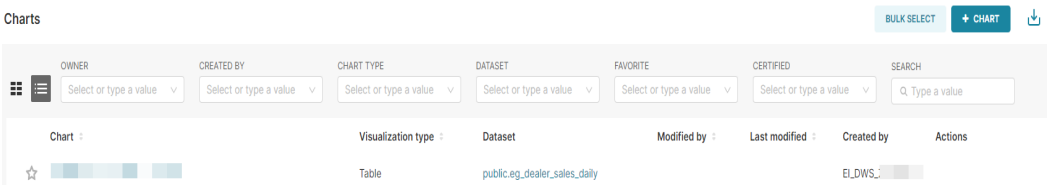


The **Save chart** panel is displayed.

Select **Save (Overwrite)** or **Save As**, and then click **Save**. The chart name is filled by default and can be modified. **Save (Overwrite)** will overwrite the current chart settings, and **Save As** will save the chart as a new one.



After the chart is saved, you can view the new chart on the chart list page.



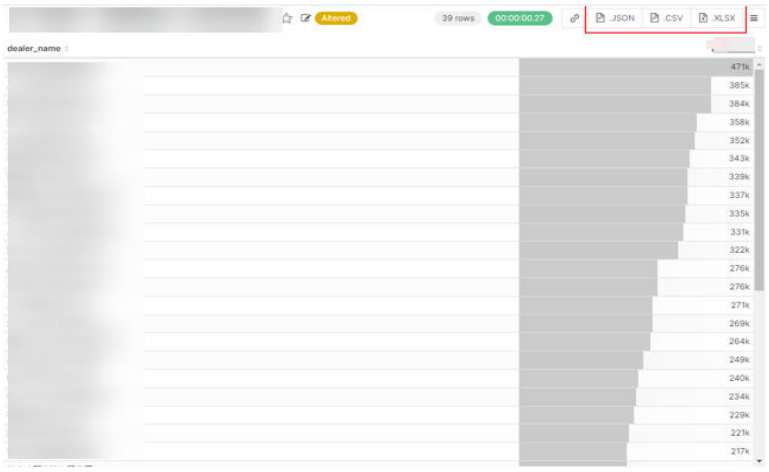
The chart is created. Next, we will show you how to organize your charts and create your own dashboard.

Exporting Chart Data

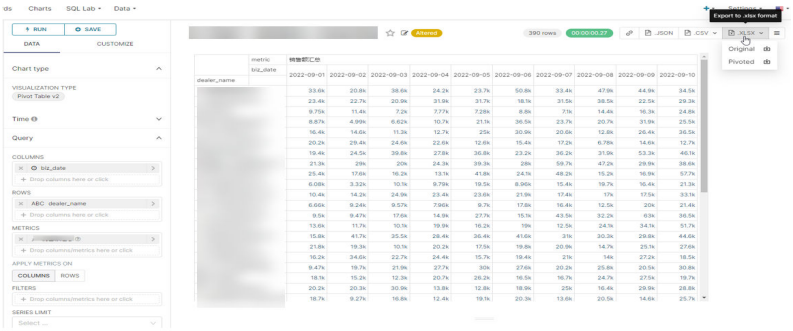
The source data in the chart can be exported to a local file in JSON, CSV, or XLSX format.

Step 1 Go to the page of the chart you want to export.

Step 2 In the upper right corner, select an export format and click the export button to export data.



Step 3 When the chart type is **Pivot Table** or **Pivot Table v2**, the buttons corresponding to the CSV and XLSX formats change to a drop-down list box. You can select **Original** or **Pivoted** from the drop-down list box to export data.



Step 4 After you click a format, the file is downloaded to a local folder through your browser.

----End

3.5.2 Dashboards

3.5.2.1 Overview

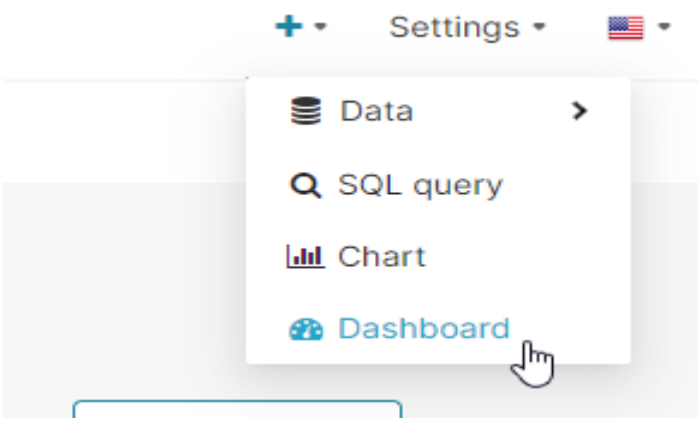
After a chart is created, you can create your own dashboard. A dashboard is a visualized representation of many groups of data. In QueryBoost, a dashboard is a collection of charts. A dashboard can tell a story by combining different types of charts. If used properly, the dashboard can be a powerful tool to display data charts and monitor information based on dynamic data.

This chapter covers everything you need to know about working with dashboards, from creating and editing them to managing and filtering them.

3.5.2.2 Creating a Dashboard

You can create a dashboard in either of the following ways:

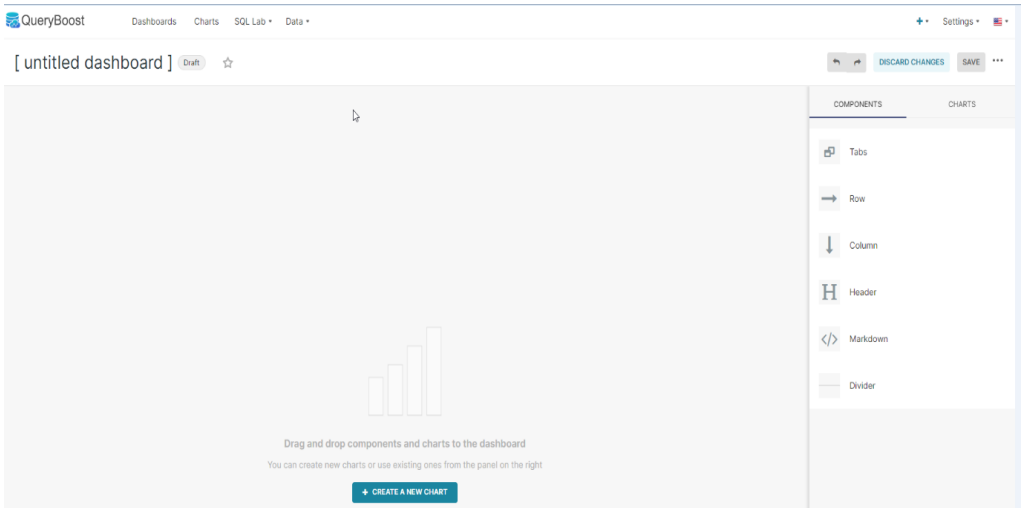
- To quickly create a dashboard, move the cursor to + in the upper right corner and click **Dashboard**.



- Click **Dashboard** in the top option bar to go to the dashboard page, and click + DASHBOARD in the upper right corner to create a dashboard.

3.5.2.3 Editing a Dashboard

In the dashboard list, click the dashboard title to go to the dashboard details page. Click in the upper right corner to edit the dashboard.



Editable items of a dashboard include:

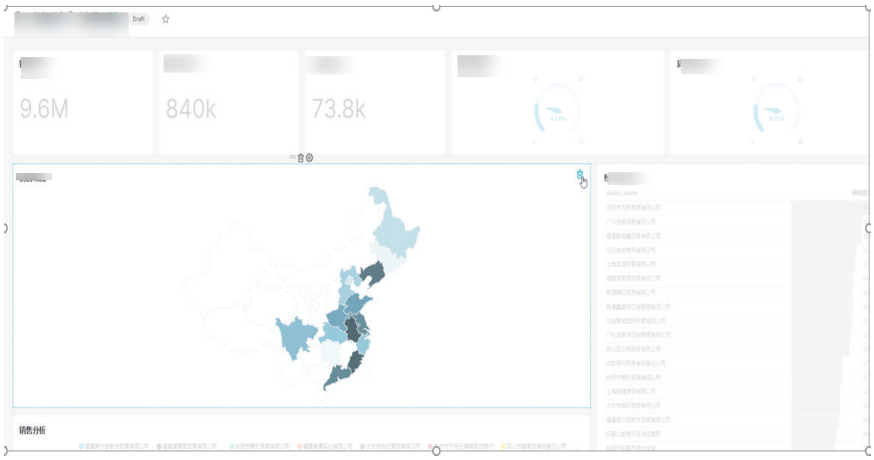
- **Dashboard title**: title of the current dashboard. A newly created dashboard is named **Unnamed Dashboard** by default, which you can click to edit the title.
- **Dashboard status**: The default value is **Draft**. In this status, only the Admin role and authorized users can viw it. Click to change the status to **Published** to publish it to all authorized users.
- **Add to favorites**: Click the pentagon to add the dashboard to, or remove it from your favorites.
- **Cancel editing**: Cancel the editing and exit the editing mode.
- **Save**: Save the editing and exit the editing mode.

Components and Charts

Available components and charts are displayed on the right of the dashboard. You can drag them to the dashboard. You can also adjust the position and size of each chart component on the dashboard. All charts of the current user are displayed on the Charts tab page. You can search for a specific chart by keyword and sort the charts.

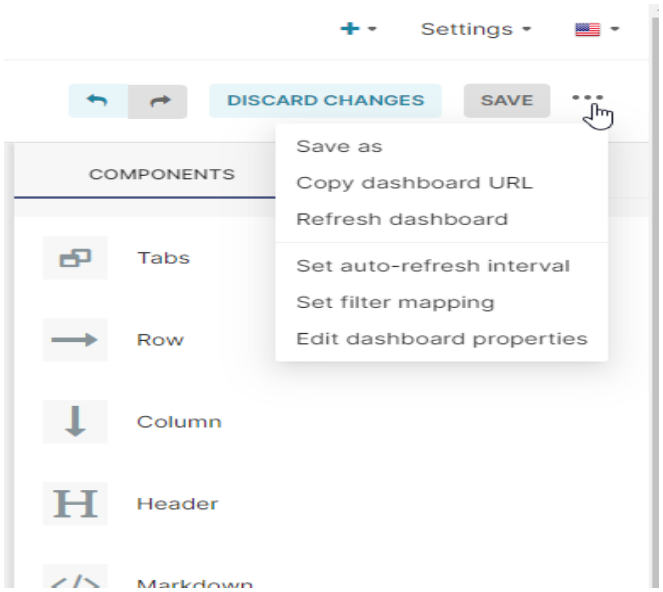
Removing a Chart

In the editing mode of the dashboard, click on the up right corner of the chart to remove the chart. This operation only removes the reference oto the chart, but does not delete the chart.

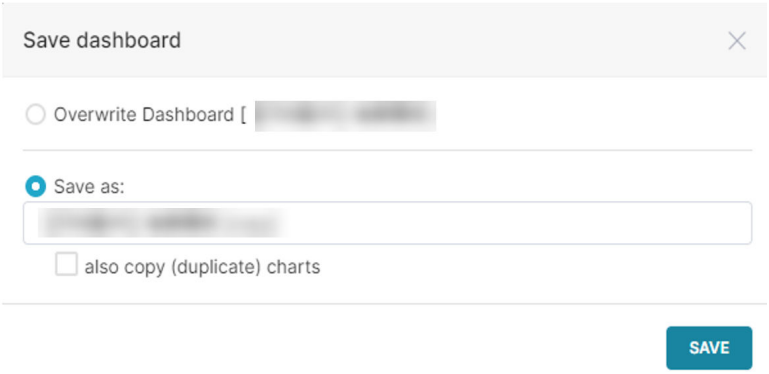


Dashboard Options

In editing mode, click the ••• icon in the upper right corner of the page. The following options are displayed:



- **Save As**: After you click **Save As**, the **Save dashboard** dialog box is displayed. By default, the current chart is referenced. (If you select **also copy (duplicate) charts**, a backup of the chart in the dashboard is created and the new chart is referenced. Exercise caution when selecting this option.)



- **Copy dashboard URL**: Copy the current dashboard URL and share it with others.
- **Refresh dashboard**: Forcibly refresh the current dashboard data.
- **Set auto-refresh interval**: Set the automatic refresh interval of the dashboard. The options are seconds (10 or 30), minutes (1, 5, or 30), or hours (1, 6, 12, or 24). This setting applies only to the current session.
- **Set permanent auto-refresh**: If you want to set permanent automatic refresh, you need to modify the JSON metadata in the dashboard properties. Modify or add the **refresh_frequency** parameter to set the refresh interval (in seconds). To set the refresh interval to one hour, set the **refresh_frequency** parameter to **3600**.
- **Set filter mapping**: Enable the filter ranges. This option must be first configured.
- **Edit dashboard properties**: Edit other attributes such as the owner of the dashboard. For details, see **Editing Dashboard Properties**.

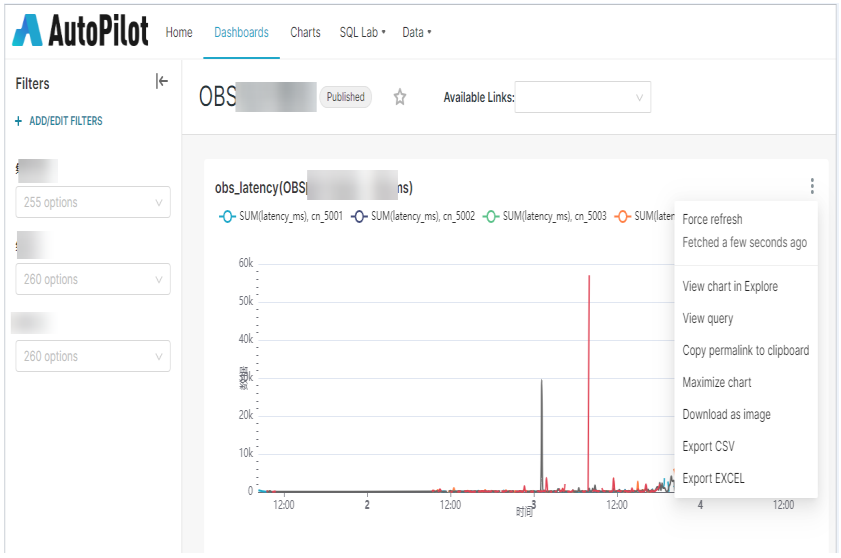
NOTE

Other options available when exit the editing mode include:

- **Download as Image**: Download the image of the entire dashboard in JPEG format.
- **Full-screen**: Displays the dashboard in full-screen without the toolbar on the top. This function is used together with the full-screen function of the browser. Select **Exit full screen** to return to the default view.

Chart Options

When you view a dashboard (non-editing mode), you can select the vertical ellipsis icon in the chart to display specific options of the chart.



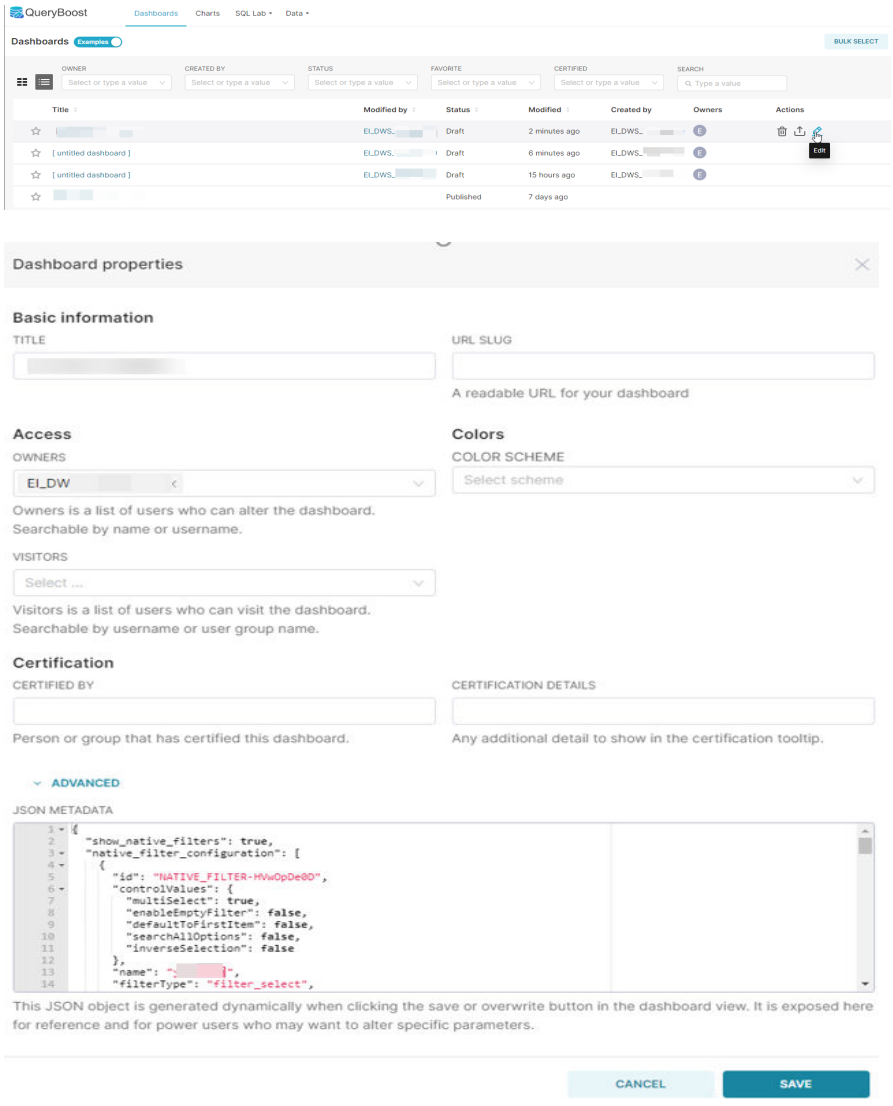
- **Refresh**: Forcibly refresh chart data.
- **View chart**: Go to the **Chart Exploration** page.
- **Check query**: The **Check query** page is displayed, showing the SQL queries of the current chart. Click  to copy the SQL statements to the clipboard.

- **Copy chart URL**: Copy the link of the chart, which can be used for sharing. The URL directs to the chart on the dashboard.
- **Maximize chart**: View the current chart in full screen mode. Click to return to the default view.
- **Download as image**: Download the chart image in JPEG format.
- **Export CSV**: Download the chart data in CSV format.

3.5.2.4 Managing Dashboards

Editing Dashboard Properties

On the **Dashboards** page, click . The **Dashboard properties** page is displayed.

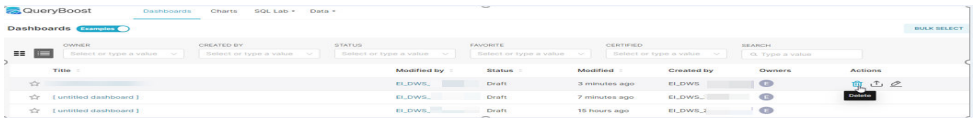


- **TITLE**: title of the dashboard.
- **URL SLUG**: Set the end of the URL to a name that is easier to remember.
- **OWNERS**: Users assigned permission to access to the dashboard.

- **COLOR SCHEME**: Color scheme of the dashboard.
- CERTIFIED BY: If this dashboard is set to **Certified**, fill in the person or group that has certified this dashboard. After certification,  is displayed before the dashboard title.
- **CERTIFICATION DETAILS**: Any other details to be displayed in the certification message.
- **JSON METADATA**: Click **ADVANCED**, the JSON metadata panel is displayed. This option is for advanced users to modify specific dashboard parameters. Generally, no modification is required.

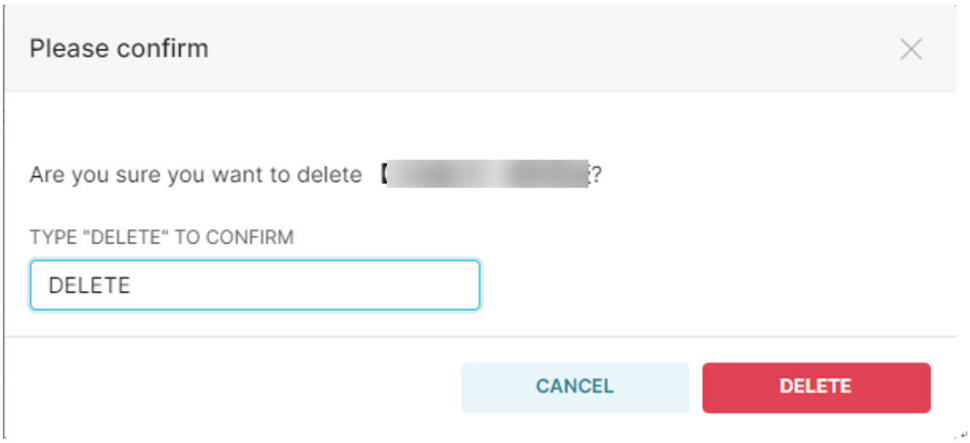
Deleting a Dashboard

Step 1 Click  to delete a dashboard. The deletion confirmation page is displayed. The delete button is unavailable by default.



Step 2 Manually enter **Delete** in the text box to confirm the deletion.

Step 3 After entering "DELETE", click **DELETE** to delete the dashboard.



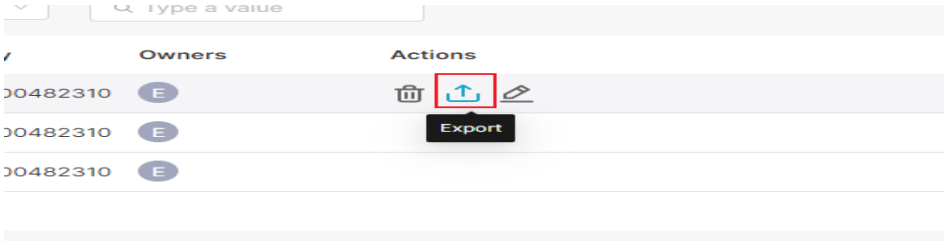
NOTE

The dashboard cannot be retrieved after being deleted. Exercise caution when performing this operation.

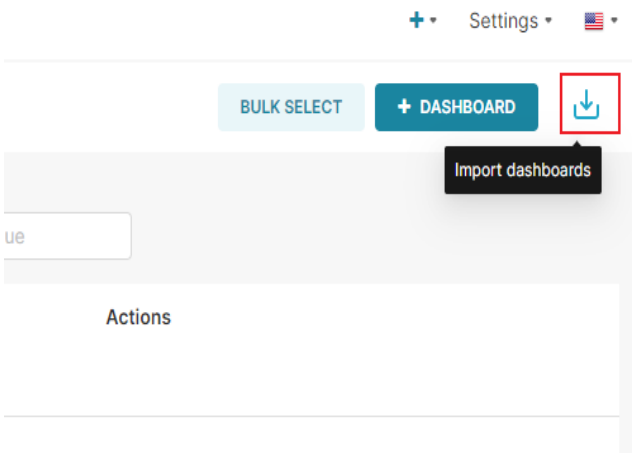
----End

Importing and Exporting a Dashboard

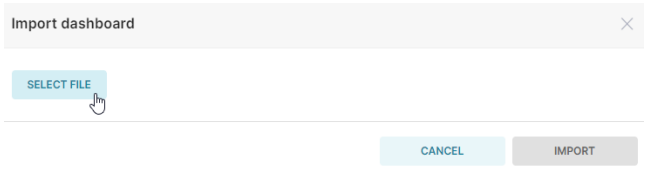
- **Export**: The developed dashboard supports can be exported. Go to the dashboard list, find the dashboard to be exported, and click **Export** in the **Actions** column (you must have the export permission). The browser automatically downloads the JSON file to the local directory (the exported content does not contain dashboard data).



- **Import**: To import a dashboard, click the import button in the upper right corner of the dashboard home page.

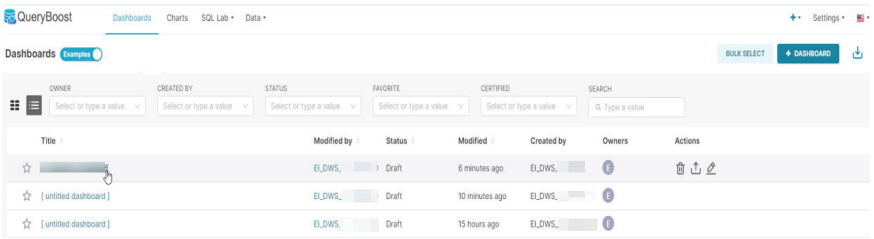


In the displayed **Import dashboard** dialog box, click **SELECT FILE**, select the file to be uploaded, and click **IMPORT**. A dashboard with the same name as that before the export is generated. (The prerequisite is that the same data source exists. Otherwise, an error is reported.)



3.5.2.5 Dashboard Filtering

The dashboard filtering function can be used for all charts on a dashboard to find the correlation of data in different charts. To go to the dashboard details page and use the filtering function of the dashboard, click **Dashboard** on the toolbar. On the displayed page, click a dashboard title, and click  on the left of the screen to expand the filters.



NOTE

To retract the sidebar, select the  icon.

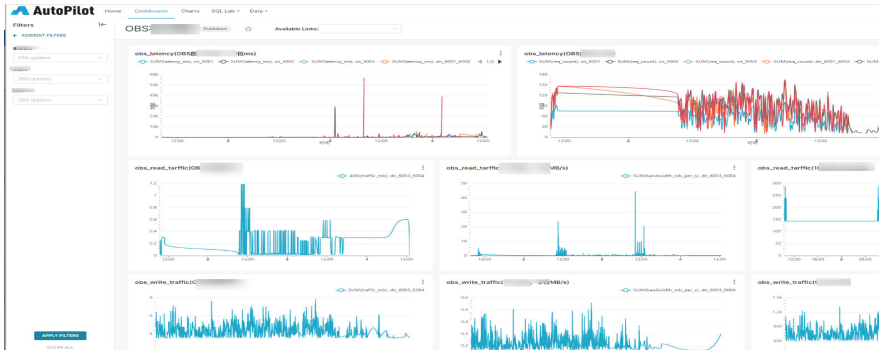
Adding Filters

Step 1 Click the icon in the filter component. The **Add and edit filters** page is displayed. You can add a filter and set its application scope.

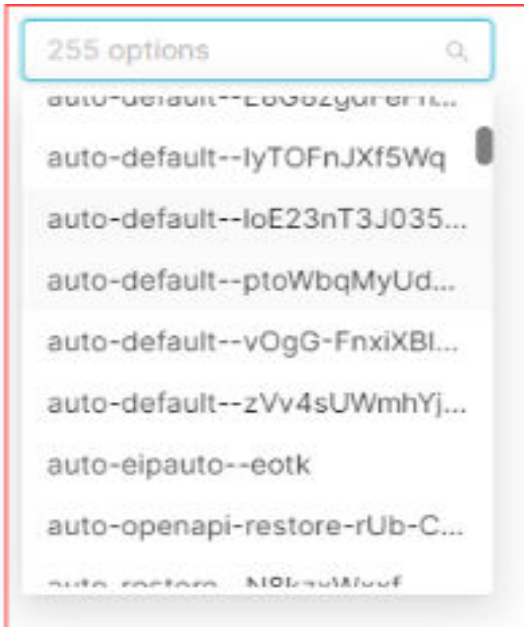
Step 2 The configuration items of the filter are as follows:

- **FILTER TYPE**: Generally, it is the data type of the filter column, which can be value, value range, time range, time, or time granularity.
- **FILTER NAME**: name of the filter. It is displayed on the dashboard page and filter component. The filter name can be the same as the filter column.
- **DATASET**: data set whose columns need to be filtered.
- **COLUMN**: filter column.

Step 3 After setting the preceding configuration items, click **Save**. A filter is added. When you move the pointer to a filter item, the chart associated with the selected data set in the filter item is highlighted.



Step 4 Select **1** and click **Apply**.



Step 5 The filter is applied to related charts, and filtered data is displayed. Filters are displayed in the upper right corner of the chart.

----End

Other Filter Configurations

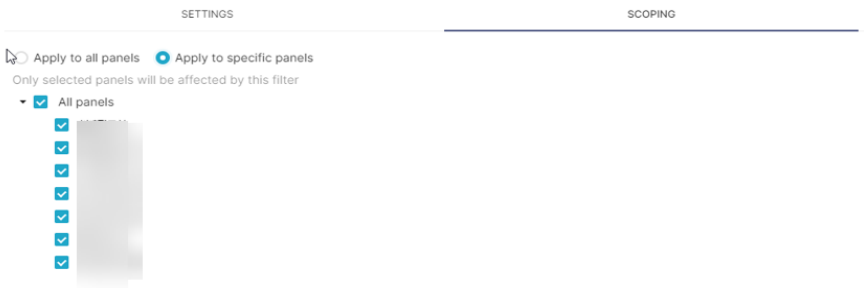
In addition to mandatory configuration items, the filter provides other configuration items, including basic and advanced configuration items. The following use the value type filter as an example.

- **Filter has a default value**: By default, no default value is required for the filter. If you select this option, you need to select a default value. The default filter will be used on the dashboard.
  - **Multi-value**: This option is selected by default. You can select multiple values.
  - **Mandatory**: If this parameter is selected, a filter value must be selected. **Filter has a default value** is enabled by default.
- **Hierarchical Filter**: If this option is selected, you can add a parent filter for the current filter. To use the cascading filtering function, you need to set a parent filter for each child filter. For example, if you have a monthly filter, you can make its parent filter a quarter. This option is only available when you have more than one filter on the dashboard.
- **First by default**: If this option is selected, the first item in the filter column is selected as the default value. If this parameter is selected, **Filter has a default value** cannot be selected.
- **Invert**: If this option is selected, the selected value is excluded.
- **Search for all filter options**: By default, a maximum of 1000 options can be loaded for each filter on the initial page. Select this box if you have more than 1000 filter values and want to enable dynamic search to load filter values as you type (which may add pressure to the database).
- **Pre-filter available value**: If this option is selected, you need to set pre-filter criteria first. When this filter is applied, the pre-filtering items will be used first.
- **Sort filter values**: If you select this option, the filter values are sorted based on the selected counters.

Filter Application Scope

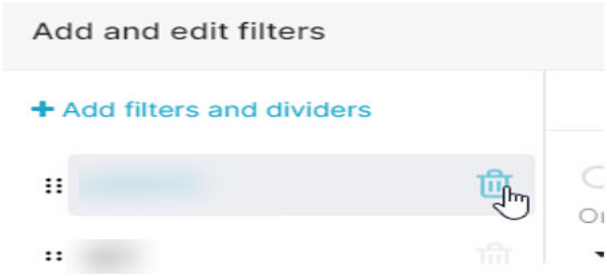
On the **Add and edit filler** page, click the **Scoping** tab. The application scope page is displayed. By default, the filter applies to all panels, that is, the entire dashboard page.

You can select **Apply to specific panels**.



Deleting a Filter

If a filter is no longer used, click  to delete it.



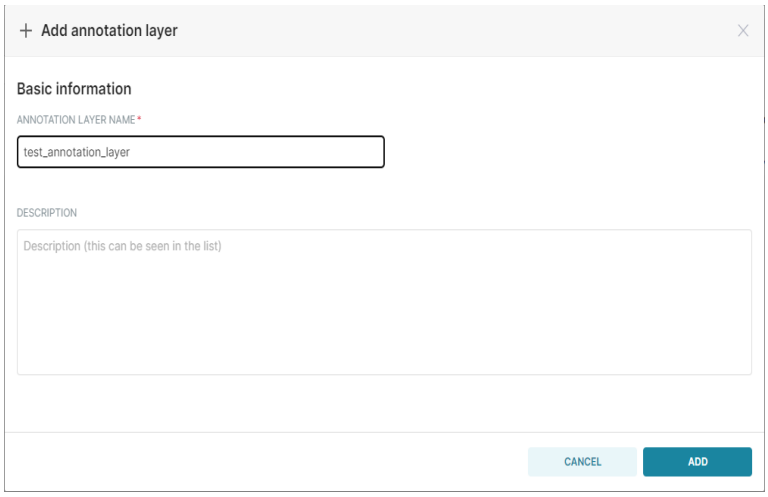
If a filter is deleted by mistake, you can click **Undo** or **RESTORE FILTER** to cancel the deletion.

3.5.2.6 Annotation Layer

The annotation layer provides users with operations such as auxiliary line setting and time area division. It is used to improve the functions of the dashboard module. You can set the annotation layer information to divide time zones in advance. The procedure is as follows:

Step 1 In the **Settings** drop-down list in the upper right corner of the home page, click **Annotation Layer**. The **Annotation Layer** page is displayed.

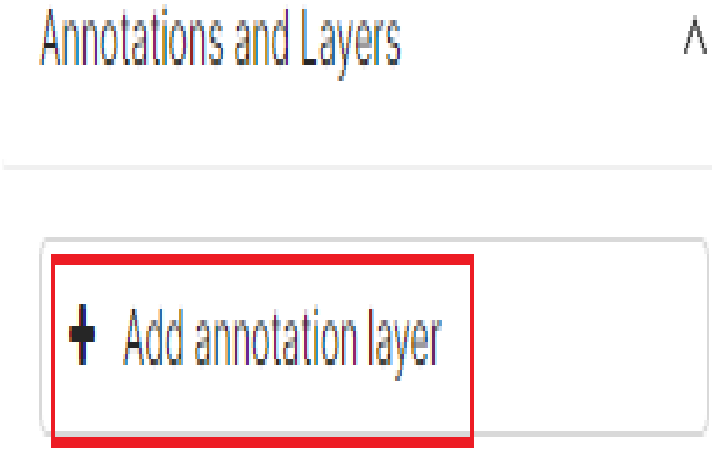
Step 2 Click **+ Annotation Layer** in the upper right corner to create an annotation layer.



Step 3 After you click **ADD**, the annotation layer is created and the information page of the annotation layer is displayed.

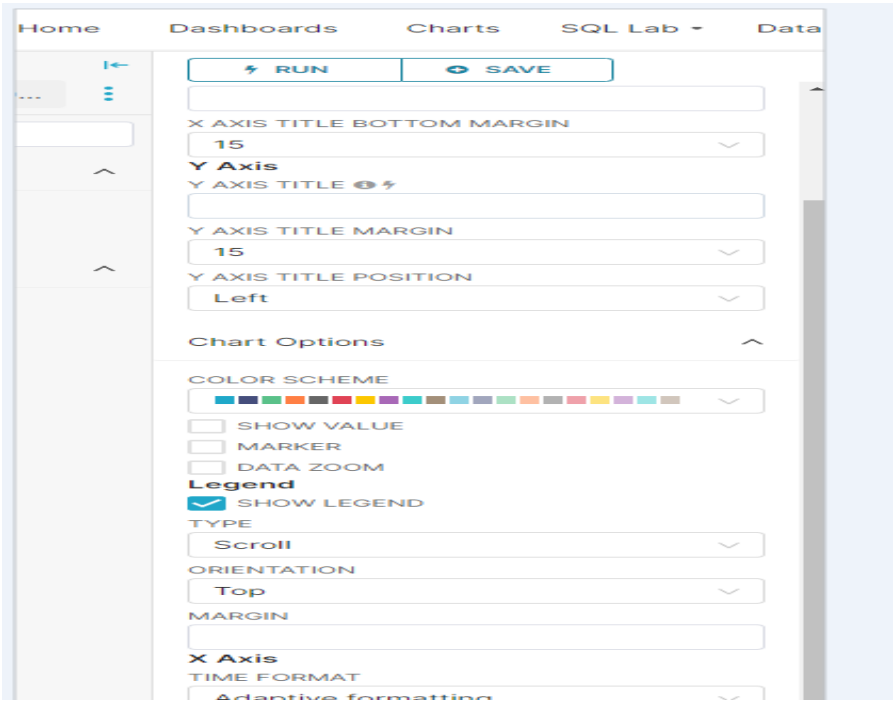
Step 4 Click **+ Annotation** to access the comment adding form. Enter the name and date as required. Then the annotation can be used on the dashboard.

Step 5 Open a chart that supports the annotation layer. The annotation and annotation layer toolbar is displayed in the data bar. Click **Add annotation layer**.

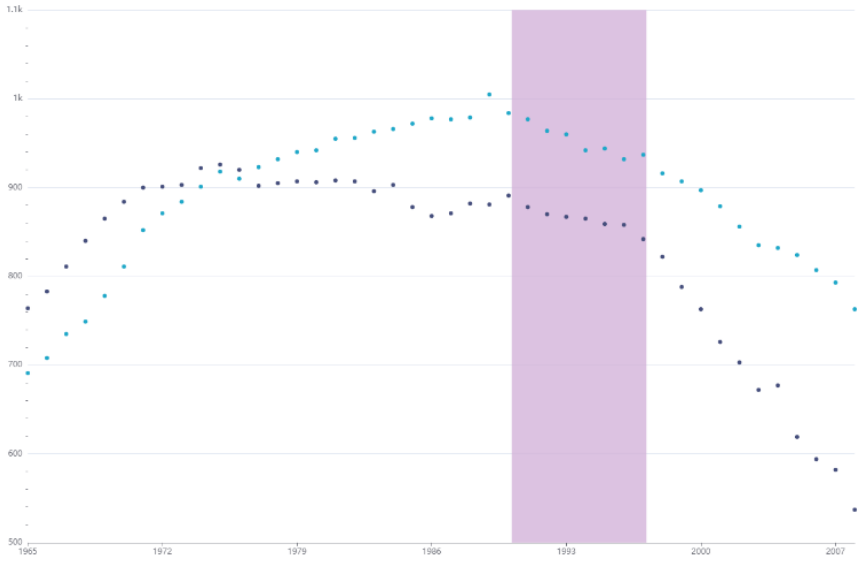


Step 6 The annotation layer configuration page is displayed.

- **Name**: name of the annotation layer displayed in the chart configuration.
- **Hide Layer**: Hide the annotation layer in the chart.
- **Annotation Layer Type**: format of the annotation layer.
- **Formula/Annotation Source**: Configure the annotation layer content based on the selected type.
- **Style**: annotation auxiliary line style.
- **Opacity**: Opacity of the annotation auxiliary line.
- **Color**: color of the annotation auxiliary line.
- **Line Width**: line width of the annotation auxiliary line.



Step 7 If you select **Event** and **Interval** from the **Annotation Layer Type** drop-down list box, you can find the created annotation in the **Source** area. After the configuration is complete, the style is as follows:



----End

3.6 Permissions

3.6.1 First Login

Log in to QueryBoost as a Cloudscope user. Before using QueryBoost, register a Cloudscope user. You are advised to use the Cloudscope account to log in to the system for the first time. After the login is successful, the master account is assigned the administrator role and has all control rights of the system.

3.6.2 Permission Management

3.6.2.1 Overview

QueryBoost uses role-based permission control and resource-based permission control.

The system has three built-in roles:

- **Administrator**: system administrator who has all rights in the system.
- **Developer**:
  - Has permissions on all menus except that under **Settings** > **Security**.
  - Does not have the permission to connect to the database.
  - Has the operation permission on specified resources in accordance with the resource permission (owner/user/none).

- **Visitor**: has no permission and can only access the dashboard based on the resource permission. Resource permissions include datasets, charts, and dashboards.

The following table lists the role resource permissions.

Table 3-5 Role resource permissions

Role/ Permission	Resource Permission	Dataset Permission		Chart Permission		Dashboard Permission			
		Read	Write	Read	Write	Read (Draft)	Write (Draft)	Read (Released)	Write (Released)
Administrator	All permissions	-	-	-	-	-	-	-	-
Developer	None	×	×	×	×	×	×	×	×
	User	√	×	√	×	√	×	√	×
	Owner	√	√	√	√	√	√	√	√
Visitor	None	×	×	×	×	×	×	×	×
	User	×	×	×	×	√	×	√	×
	Owner	×	×	×	×	√	×	√	×

NOTE

- √ indicates having all permissions, × indicates having no permissions (The system administrator has all permissions by default).
- Read permission on resources:
  - Read permission on the dataset: Use the dataset to create a chart.
  - Read permission on charts: Access the chart details page and use charts on the dashboard.
  - Read permission on the dashboard: Access the dashboard page and access the chart details page based on the chart permission.
- Resource write permission: Edit and delete resources.
- Because visitors can only access the dashboard, the chart and dataset authorization does not involve the visitor role.
- A draft dashboard can be accessed by users through an url, but is not displayed in the dashboard list. If you want to display the dashboard in the list, you can add the dashboard to favorites.

3.6.2.2 Row-Level Roles

Row-level permission roles are created by users and can be used to visit different data in the same dataset.

Step 1 In the **Settings** drop-down list in the upper right corner of the home page, click **Row-level Permission**. The **Row-level Permission** page is displayed.

Step 2 Click **+** in the upper right corner. The **Add row-level security filter** page is displayed.

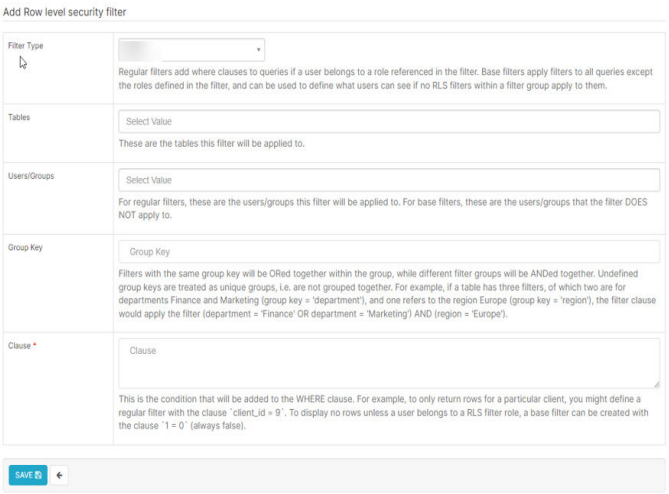


Table 3-6 Adding row-level security filtering fields

Field	Description	Example
Filter Type	Filter type: • Regular: Apply clause filtering to specified roles. • Base: Apply clause filtering to all queries except roles defined in the filter.	The default value is **Regular**.
Tables	Name of the data table to which the clause is applied, corresponding to the information about the dataset created by the user.	test_table
Users/ Groups	Set the target user group for row-level filtering (note that the filtering type affects the filter coverage).	test_user_group
Group Key	The filters with the same group key perform the **OR** operation together in the group, while the filters with different group keys perform the **AND** operation together.	test_column
Clause	Where clause.	test_column = 'test_value'

Step 3 After the configuration is complete, click **Save**. The row-level permission is generated.

----End

3.6.2.3 Examples

Row-Level Permission Application Example

In actual applications, different roles in the same department need to only access different data on the same dashboard.

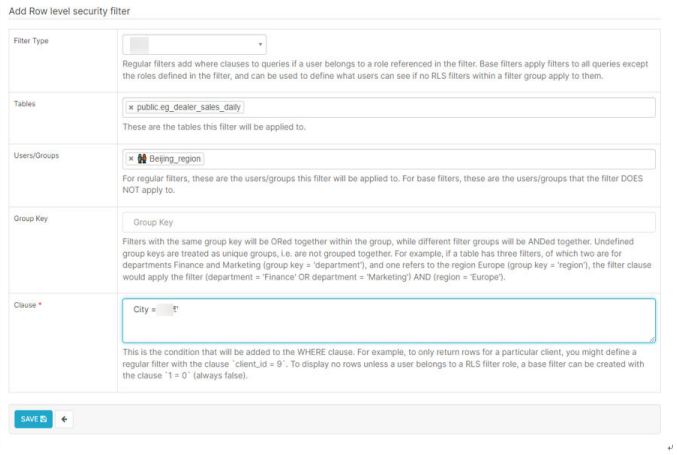
For example, if employees in the marketing department access sales orders, employees in different regions can view only the order information in their own regions. In this case, we can implement row-level permission control to user groups.

Create a test user group **region_bj**.

Step 1 Move the cursor to **Settings** on the right of the menu bar and click **Row-level Permission** in the drop-down list box. The row-level permission list page is displayed.

Step 2 Click the plus sign (+) in the upper right corner to add a permission.

- Select **sales_detail** as the data table and **region_bj** as the user group.
- Enter **region = '*region*'** in the clause.



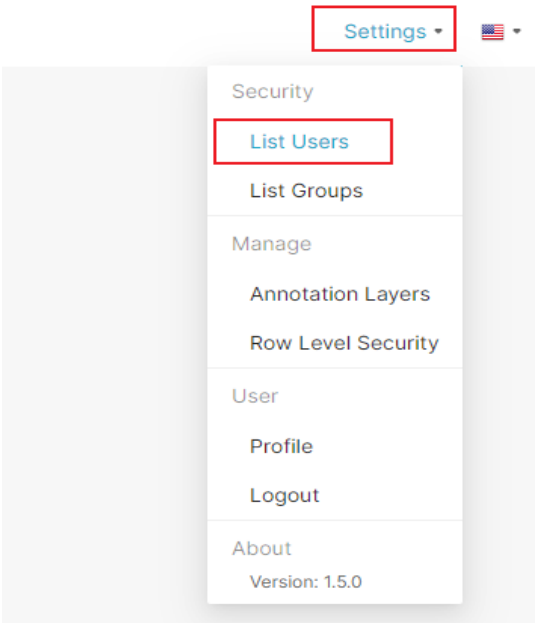
Step 3 Click **Save**.

----End

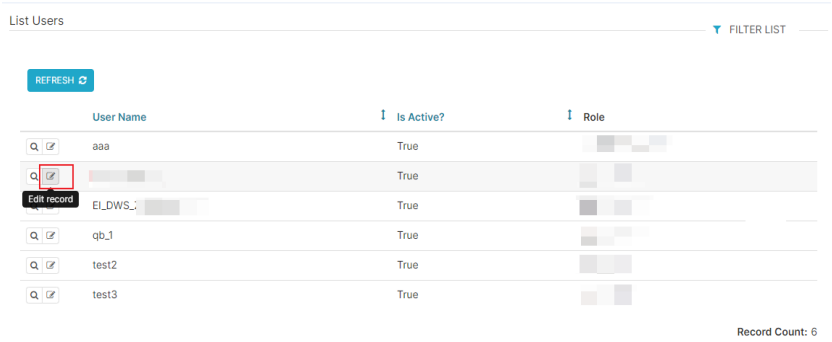
3.6.3 Deleting a User

You are not advised to delete QueryBoost users. You only need to disable QueryBoost users. After a user is disabled, the user cannot log in to the system.

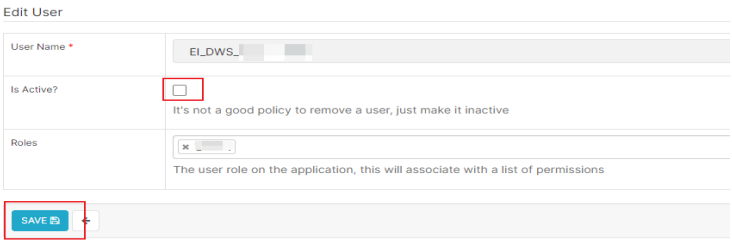
Step 1 Move the cursor to **Settings** on the right of the menu bar and click **List Users** in the drop-down list box. The user list page is displayed.



Step 2 Click **Edit User** on the left of the user to be disabled.



Step 3 Deselect the **Activate** check box, and then click **SAVE**.



----End

4 User-defined Monitoring Metrics

4.1 Overview

Overview

You define monitoring metrics and run shell and SQL commands to collect cluster or database information to the IoT database. The collected data is displayed in your configured QueryBoost charts. For detailed operations, see **Visualized Dashboard Analysis (QueryBoost)**. This function provides users with a set of metric-defining capabilities. With only a JSON file that defines metrics, you can implement the entire metric process from collecting metrics from the cluster to writing the collected data to the IoT library. You do not have to update the product version to define new metrics. This can meet the requirement of diversified and ever-changing services on the live network. In this way, you can also perform quick troubleshooting and O&M.

Example

JSON used to monitor the concurrency:

```
{
   "plugin_name": "business_concurrency",
   "namespace": "gauss.autopilot.database.query",
   "is_enable": true,
   "alarm_enable": true,
   "override": false,
   "scope": "CCN",
   "metric_table": "DWS_BUSINESS_CONCURRENCY",
   "support_version": "8.0.0",
   "plugin_version": 1681833600000,
   "type": "query",
   "plugin_type": "runScheduled",
   "collect_rate": 180,
   "res_limit": 100,
   "active_clusters": {
      "enable_all_clusters": true
   },
   "fields": [
      {
         "name": "username",
         "data_type": "string",
         "column_name": "user_name",
```



```
            "column_type": "varchar(64)"
        },
        {
            "name": "coorname",
            "data_type": "string",
            "column_name": "coorname",
            "column_type": "text",
            "is_tstag " : true
        },
        {
            "name": "state",
            "data_type": "string",
            "column_name": "state",
            "column_type": "text"
        },
        {
            "name": "count",
            "data_type": "integer",
            "column_name": "count",
            "column_type": "integer"
        }
    ],
    "query": {
        "query_sql":
"c2VsZWN0IHVzZW5hbWUsIGNvb3JuYW1lLCBzdGF0ZSwgY291bnQoMSkgZnJvbSBwZ3hjX3N0YXRfYWN0aXZpdHkgZ3JvdXAgYnkgY29vcm5hbWUsIHVzZW5hbWUsIHN0YXRlIG9yZGVyIGJ5IGNvdW50Ow==",
        "all_database": false,
        "databases": [
            "postgres"
        ]
    }
}
```

4.2 Metric Development Guide

Custom Metric Details

The following table lists the fields and comments of user-defined monitoring metrics.

Table 4-1 Custom metric details

Field Name	Remarks
plugin_name	Plug-in name, which is unique and can contain uppercase letters, lowercase letters, and underscores (_).
scope	Execution scope of a user-defined metric. • CN: All CNs • CCN: Only CCN. • HOST: All hosts.

Field Name	Remarks
namespace	**Prefix**: gauss.autopilot. **Collection dimensions**: ● cluster ● database ● instance ● node ● business **Collection language (corresponding to the Type field)**: ● **query**: SQL statement. ● **shell**: shell script. Example: gauss.autopilot.cluster.query
is_enable	Whether a metric is enabled. ● **true**: Enabled. ● **false**: Disabled.
alarm_enable	Specifies whether alarms need to be configured for the metric. ● **true**: Alarms need to be configured. ● **false**: Alarms do not need to be configured.
metric_table	Name of the table to which the collected data is imported. The value must be unique and can contain uppercase letters, lowercase letters, and underscores (_).
support_version	Earliest cluster version supported. For example, 8.2.0 indicates that all clusters of version 8.2.0 or later are supported.
type	Language type of the script to be executed. ● **query**: SQL statement. ● **shell**: shell script.
plugin_type	Metric execution type. ● **runScheduled**: The task is executed periodically. ● **runOnce**: The task is executed only once.
collect_rate	Frequency of periodic execution, in seconds. The minimum value is **15**. For example, **60** indicates that the collection script is executed every 60 seconds.
collect_range	The collection time range is specified as a timestamp accurate to the second. If this parameter is not set, data will be collected continuously. The format is **[***Start time***,***End time***]**, for example, **[1692929703,1692929903]**.

Field Name	Remarks	
res_limit	Maximum number of data rows that can be collected each time. The minimum value is **1** and maximum value is **1000**. For example, **100** indicates that only 100 pieces of data are collected each time.	
active_clusters	enable_all_clusters	Whether all clusters are supported. ● **true**: All clusters are supported. ● **false**: Not all clusters are supported.
	clusters	Cluster ID array. This parameter is valid only when **enable_all_clusters** is set to **false**. It is used to create the available cluster list. For example, **["f1f00169-4383-41e2-99b1-bc1cff492824","86caa385-7d6c-411d-be0d-c5535c490761"]**.
invalid_clusters	Cluster ID array. If this parameter is not specified, the value is empty by default. Indicates the cluster that is not allowed to use the collection metric. For example, **["f1f00169-4383-41e2-99b1-bc1cff492824","86caa385-7d6c-411d-be0d-c5535c490761"]**.	
fields	name	Name of a column in the collected information.
	data_type	Type of a column in the collected information.
	column_name	Name of a column in the **metric_table** table.
	column_type	Name of a column in the **metric_table** table.
	is_tstag	Indicates whether to partition data based on this field. If this parameter is not specified, the default value **false** is used. ● **true**: Enable data partitioning. ● **false**: Disable data partitioning.
query	query_sql	This parameter is valid only when **type** is set to **query**. Indicates a SQL statement, which must be encoded using Base64.
	all_database	This parameter is valid only when **type** is set to **query**. It indicates whether to execute the SQL statement in all databases. ● **true**: Execute the SQL statement in all databases. ● **false**: Not to execute the SQL statement in all databases.
	databases	This parameter is valid only when **type** is set to **query** and **all_database** is set to **false**. It specifies the list of execution databases. The value format is **["dbname1","dbname2"]**.

Field Name	Remarks
cmd	This parameter is valid only when **type** is set to **shell**. The shell script must be encoded using **Base64**.
plugin_version	Metric version, which is a millisecond-level timestamp. If this parameter is not specified, the current timestamp is used.
override	This parameter is not used currently and does not need to be added. ● **true**: Yes. ● **false**: No.
create_time	Creation time, which is set by the program and does not need to be added. Milliseconds-level timestamp.
publish_state	Metric publish status, which is processed by the program and does not need to be added.

Example

- The following is an example of a user-defined SQL metric:

```
{
  "plugin_name":"business_obs_write_traffic",
  "namespace":"gauss.autopilot.database.query",
  "is_enable":true,"scope":"CCN",
  "metric_table":"DWS_BUSINESS_OBS_WRITE_TRAFFIC",
  "support_version":"9.0.0",
  "type":"query",
  "plugin_type":"runScheduled",
  "collect_rate":60,
  "res_limit":1000,
  "active_clusters":{
     "enable_all_clusters":true
  },
  "fields": [
     {
         "name":"nodename",
         "data_type":"string",
         "column_name":"node_name",
         "column_type":"TEXT"
     },
     {
         "name":"traffic_mb",
         "data_type":"string",
         "column_name":"traffic_mb",
         "column_type":"FLOAT"
     },
     {
         "name":"bandwidth_mb_per_s",
         "data_type":"string",
         "column_name":"bandwidth_mb_per_s",
         "column_type":"FLOAT"
     },
     {
         "name":"reqcount",
         "data_type":"long",
         "column_name":"req_count",
         "column_type":"BIGINT"
     },
     {
         "name":"logtime",
```



```
            "data_type":"string",
            "column_name":"log_time",
            "column_type":"TIMESTAMP WITH TIME ZONE"
        }
    ],
    "query":{
        "query_sql":
"c2VsZWN0IGdzX2NyZWF0ZV9sb2dfaW50ZXJuYWxfdGFibGVzKCkgd2hlcmUgKHNlbGVjdCBjb3VudCgq
KSBxdWVyeV9udW0gZnJvbSBwZ19jYXRhbG9nLnBnX2NsYXNzIGNscywgcGdfY2F0YWxvZy5wZ19uYW1lc
3BhY2UgbnNwIHdoZXJlIGNscy5yZWxuYW1lID0gJ2dzX3Byb2ZpbGVfbG9nX2ludGVybmFsX2Z0JyBhbmQ
gY2xzLnJlbG5hbWVzcGFjZSA9IG5zcC5vaWQgYW5kIG5zcC5uc3BuYW1lID0gJ3B1YmxpYycpID0gMDsKd2
l0aCBvYnNfd3JpdGVfdHJhZmZpYyBhcyAoc2VsZWN0IG5vZGVuYW1lLCBzdW0oc2l6ZSkgYXMgc2l6ZSBv
xMDI0IGFzIHRyYWZmaWMsIChzdW0ocmVxc2l6ZSkvMTAyNC8xMDI0KS8oc3VtKHJlcXVzZWMpLzEwM
DAvMTAwMCkgYXMgYmFuZHdpZHRoLCBzdW0ocmVxY291bnQpIGFzIGNvdW50LCB0b190aW1lc3RhbX
AoZmxvb3IoZXh0cmFjdChlcG9jaCBmcm9tIGxvZ3RpbWUpLzYwKSo2MCkgYXMgdGltZWludGVydmFsIGZ
yb20gZ3NfcHJvZmlsZV9sb2dfaW50ZXJuYWxfZnQgd2hlcmUgcmVxc3JjPSdvYnMnIGFuZCByZXF0eXBlPSd
3cml0ZScgYW5kIHRpbWVpbnRlcnZhbCBiZXR3ZWVuIHRvX3RpbWVzdGFtcChmbG9vcihleHRyYWN0KGV
wb2NoIGZyb20gZ3NfcHJvZmlsZV9sb2dfaW50ZXJuYWxfZnQgd2hlcmUgcmVxc3JjPSdvYnMnIGFuZCByZXF0eXBlPSd
wgJzVNJykpLzYwKSo2MCkgZ3JvdXAgYnkgbm9kZW5hbWUsIHRpbWVpbnRlcnZhbCBvcmRlciBieSBub2Rlb
mFtZSwgdGltZWludGVydmFsKQpzZWxlY3Qgbi5ub2RlX25hbWUgbm9kZW5hbWUsIGNvYWxlc2NlKG8u
dHJhZmZpYywgMCkgdHJhZmZpYywgY29hbGVzY2Uoby5iYW5kd2lkdGgsMCkgYmFuZHdpZHRoLCBjb2F
sZXNjZShvLmNvdW50LDApIGNvdW50LCBjb2FsZXNjZShvLnRpbWVpbnRlcnZhbCx0b190aW1lc3RhbXAo
Zmxvb3IoZXh0cmFjdChlcG9jaCBmcm9tIChzZWxlY2Qgc3lzZGF0ZSAtIGludGVydmFsICc1TScpKS82MCkq
NjApKSBsb2d0aW1lIGZyb20gcGd4Y19ub2RlIG4gbGVmdCBqb2luIG9ic193cml0ZV90cmFmZmljIG8gb24g
bi5ub2RlX25hbWUgPSBvLm5vZGVuYW1lOw==",
        "all_database":false,
        "databases":[
            "postgres"
        ]
    }
}
```

- Example of user-defined metrics defined by shell script:

```
{
  "plugin_name":"database_cu_check",
  "namespace":"gauss.autopilot.database.query",
  "is_enable":true,"scope":"CCN",
  "metric_table":"DWS_DATABASE_TABLES_CU_CHECK",
  "support_version":"8.2.0",
  "plugin_version":1681833600000,
  "type":"shell",
  "plugin_type":"runScheduled",
  "collect_rate":14400,
  "res_limit":200,
  "active_clusters":{
     "enable_all_clusters":true
  },
  "fields":[
     {
        "name":"database",
        "data_type":"string",
        "column_name":"database",
        "column_type":"text"
     },
     {
        "name":"schema",
        "data_type":"string",
        "column_name":"schema",
        "column_type":"text"
     },
     {
        "name":"tablename",
        "data_type":"string",
        "column_name":"tablename",
        "column_type":"text"
     },
     {
        "name":"tablesize",
        "data_type":"string",
```



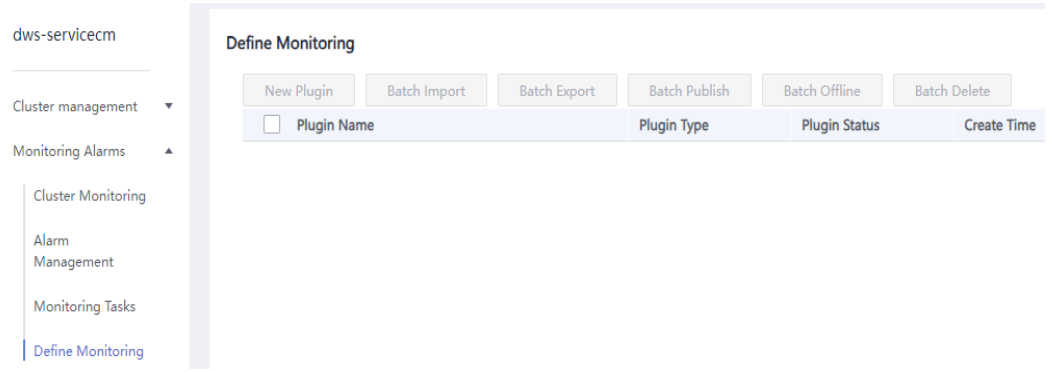
```
        "column_name":"tablesize",
        "column_type":"BIGINT"
      },
      {
        "name":"rows_per_cu",
        "data_type":"string",
        "column_name":"rows_per_cu",
        "column_type":"BIGINT"
      }
   ],
   "cmd":
"IyEvYmluL2Jhc2gKCmV4cG9ydCBDTlBPUlQ9ODAwMApleHBvcnQgcmVzQ291bnQ9MAoKVElUTEU9KC
JkYXRhYmFzZSIgInNjaGVtYSIgInRhYmxlbmFtZSIgInRhYmxlc2l6ZSIgInJvd3NfcGVyX2N1IikKZWNobyAke1
RJVExFW0BdfQoKZnVuY3Rpb24gZ2V0Q05Qb3J0KCkKewogICAgcG9ydFRtcD1gZWNobyAiJHtMT0NBTF
9JTlNUfSIgfGdyZXAgQ058YXdrIC1GJ3wnICd7cHJpbnQgJDV9J2AKICAgIGlmIFsgJHBvcnRUbXAgLWdlIDE
wMDAgXSAmJiBbICRwb3J0VG1wIC1sZSA2NTUzNSBdOyB0aGVuCiAgICAgICAgQ05QT1JUPSRwb3J0VG1
wCiAgICBmaQp9CgpmdW5jdGlvbiBjaGVja0NDTigpCnsKICAgIHNxbD0ic2VsZWN0IHBneGNfbm9kZV9zd
HIoKSxub2RlX25hbWUgZnJvbSBwZ3hjX25vZGUgd2hlcmUgbm9kZWlzX2NlbnRyYWw9dHJ1ZTsiCiAgICB
yZXM9YGdzcWwgcG9zdGdyZXNxbDovLzoke0NOUE9SVH0vcG9zdGdyZXM/
YXBwbGljYXRpb25fbmFtZT0nT00nIC1jICIkc3FsIiAtdCAtQSAyPiYxYAogICAgZXJyPWBlY2hvICIkcmVzIiB8Z
3JlcCAiRVJST1I6Inx3YyAtbGAKICAgIGlmIFsgJGVyciAtZXEgMSBdOyB0aGVuCiAgICAgICAgZXhpdCAwCiA
gICBmaQogICAgbXljbj1gZWNobyAiJHJlcyIgfGF3ayAtRid8JyAne3ByaW50ICQxfSdgCiAgICBjY249YGVjaG
8gIiRyZXMiIHxhd2sgLUYnfCcgJ3twcmludCAkMn0nYAogICAgaWYgWyAkbXljbiAhPSAkY2NuIF07IHRoZW
4KICAgICAgICBleGl0IDAKICAgIGZpCn0KCmZ1bmN0aW9uIHBhcnNlT25lVGFibGUoKQp7CiAgICB0YWJsZ
VRtcD0iJDEiCiAgICBzY2hlbWE9YGVjaG8gIiR0YWJsZVRtcCIgfGF3ayAne3ByaW50ICQyfSdgCiAgICB0YWJs
ZT1gZWNobyAiJHRhYmxlVG1wIiB8YXdrICd7cHJpbnQgJDN9J2AKICAgIHNxbD0ic2V0IHN0YXRlbWVudF9
0aW1lb3V0PTEwMDAwO3NlbGVjdCBjYXNlIHdoZW4gKHRfY3VfY250ID0gMCBvciByZWx0dXBsZXMgPSA
wKSB0aGVuIDAgZWxzZSAocmVsdHVwbGVzLyh0X2N1X2NudC9yZWxuYXR0cykpOjpudW1lcmljKDEwKS
BlbmQgYXZnX2N1X3Jvd3MgZnJvbSBwZ19jbGFzcyBjLHBnX25hbWVzcGFjZSBuLChzZWxlY3Qgc3VtKHRv
dGFsX2N1X2NvdW50KSB0X2N1X2NudCBmcm9tIGdldF9jb2xfY3VfaW5mbygnJHtzY2hlbWF9JywnJHt0Y
WJsZX0nLDIwMCwxMDApKSBjdV9pbmZvIHdoZXJlIG4ub2lkID0gYy5yZWxuYW1lc3BhY2UgYW5kIGMuc
mVsbmFtZSA9ICcke3RhYmxlfScgYW5kIG4ubnNwbmFtZSA9ICcke3NjaGVtYX0nOyIKICAgIHJlcz1gZ3Nxb
CBwb3N0Z3Jlc3FsOi8vOiR7Q05QT1JUfS8ke2RiTmFtZX0/
YXBwbGljYXRpb25fbmFtZT0nT00nIC1jICIkc3FsIiAtdCAtQSAyPiYxYAogICAgZXJyPWBlY2hvICIkcmVzIiB8Z
3JlcCAiRVJST1I6Inx3YyAtbGAKICAgIGlmIFsgJGVyciAtZXEgMSBdOyB0aGVuCiAgICAgICAgcmV0dXJuCiAg
ICBmaQogICAgcmVzPWBlY2hvICR7cmVzOjR9YAogICAgaWYgWyAkcmVzIC1sZSAxMDAwIF0gJiYgWyAkc
mVzIC1ndCAwIF07IHRoZW4KICAgICAgIGVjaG8gIiR0YWJsZVRtcCAkcmVzIgogICAgICAgcmVzQ291bnQ9
JCgoJHJlc0NvdW50ICsgMSkpCiAgICBmaQp9CgpmdW5jdGlvbiBwYXJzZUFsbFRhYmxlKCkKewogICAgc3F
sPSJTRUxFQ1QgY3VycmVudF9kYXRhYmFzZSgpLHMuc2NoZW1hbmFtZSxzLnRhYmxlbmFtZSxwZ19jYXR
hbG9nLnN1bShzLmRuc2l6ZSkgQVMgdG90YWxzaXplIEZST00gcGdfY2F0YWxvZy5wZ19jbGFzcyBjIElOTk
VSIEpPSU4gcGdfY2F0YWxvZy5wZ19uYW1lc3BhY2UgbiBPTiBuLm9pZCA9IGMucmVsbmFtZXNwYWNlIEl
OTkVSIEpPSU4gcGdfY2F0YWxvZy5nc190YWJsZV9kaXN0cmlidXRpb24oKSBzIE9OIHMuc2NoZW1hbmFt
ZSA9IG4ubnNwbmFtZSBBTkQgcy50YWJsZW5hbWUgPSBjLnJlbG5hbWUgSU5ORVIgSk9JTiBwZ19jYXRhb
G9nLnBneGNfY2xhc3MgeCBPTiBjLm9pZCA9IHgucGNyZWxpZCBBTkQgeC5wY2xvY2F0b3J0eXBlIElOKCdI
JywgJ04nKSBJTk5FUiBKT0lOIHBnX2NhdGFsb2cucGdfYXV0aGlkIHUgb24gdS5vaWQgPSBjLnJlbG93bmVy
IHdoZXJlIHJlbG9wdGlvbnM6OlRFWFQgbGlrZSAnJW9yaWVudGF0aW9uPWNvbHVtbiUnIEdST1VQIEJZIG
Mub2lkLHMuc2NoZW1hbmFtZSxzLnRhYmxlbmFtZSx1LnJvbG5hbWUgT1JERVIgQlkgdG90YWxzaXplIER
FU0MgbGltaXQgMjAwMDsiCiAgICBkYk5hbWU9JDEKICAgIHRhYmxlTGlzdFRtcD0iYGdzcWwgcG9zdGdyZ
XNxbDovLzoke0NOUE9SVH0vJHtkYk5hbWV9P2FwcGxpY2F0aW9uX25hbWU9J09NJyAtYyBcIiRzcWxcIiA
tdCAtQSAtRiAiICIgMj4mMWAiCiAgICByZXNDb3VudD0wCiAgICBlcnI9YGVjaG8gIiR0YWJsZUxpc3RUbXAi
IHxncmVwICJFUlJPUjoifHdjIC1sYAogICAgaWYgWyAkZXJyIC1lcSAxIF07IHRoZW4KICAgICAgICByZXR1cm
4KICAgIGZpCiAgICBlY2hvICIke3RhYmxlTGlzdFRtcH0iIHx3aGlsZSByZWFkIHRhYmxlCiAgICBkbwogICAgIC
AgIGlmIFsgJHJlc0NvdW50IC1nZSA1MCBdOyB0aGVuIGJyZWFrOyBmaQogICAgICAgIHBhcnNlT25lVGFib
GUgIiR0YWJsZSIKICAgIGRvbmUKfQoKZnVuY3Rpb24gbWFpblByb2MoKQp7CiAgICBnZXRDTlBvcnQKICA
gIGNoZWNrQ0NOCiAgICBkYkxpc3Q9KGBnc3FsIHBvc3RncmVzcWw6Ly86JHtDTlBPUlR9L3Bvc3RncmVz
P2FwcGxpY2F0aW9uX25hbWU9J09NJyAic2VsZWN0IGRhdG5hbWUgZnJvbSBwZ19kYXRhYmFzZSB3
aGVyZSBvaWQgPiAxNjM4MyBvciBkYXRuYW1lID0gJ3Bvc3RncmVzJzsiIC10YCkKICAgIGZvciBkYiBpbiAiJHt
kYkxpc3RbQF19IgogICAgZG8KICAgICAgICBwYXJzZUFsbFRhYmxlICIkZGIiCiAgICBkb25lCn0KCm1haW5Q
cm9jCg=="
}
```


4.3 Metric Management

Adding User-Defined Monitoring Plug-in

Step 1 Log in to ServiceCM.

Step 2 In the navigation pane, choose **Monitoring Alarms** > **Define Monitoring**.



Step 3 Click **Add Plugin** on the right of the page, select the JSON file of the user-defined metric, and upload the new metric.

Step 4 After the submission is successful, the **Define Monitoring** page is displayed, showing information about the custom metric.

----End

Batch Operations on User-Defined Monitoring Metrics

Step 1 Log in to ServiceCM.

Step 2 In the navigation pane, choose **Monitoring Alarms** > **Define Monitoring**. You can import, export, publish, bring offline, and delete user-defined metrics in batches. The procedure is as follows:

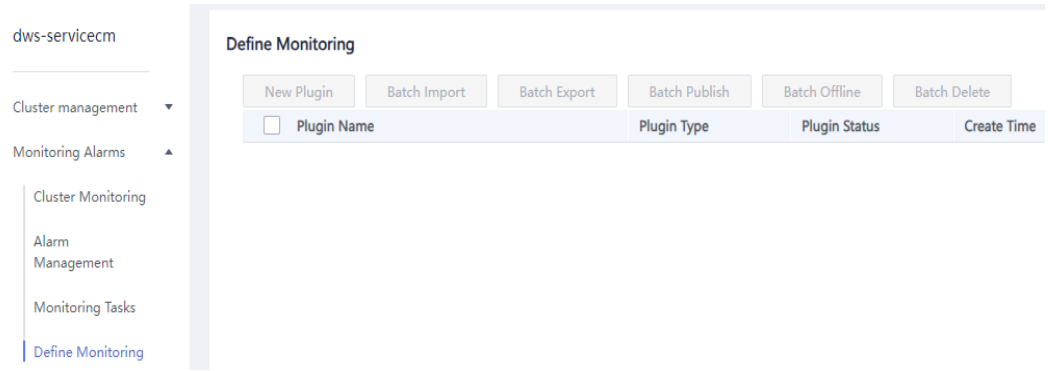


Table 4-2 Batch operations on user-defined monitoring metrics

Operation	Description
Batch import	You can upload the JSON files of multiple user-defined metrics at a time.
Batch export	You can export the JSON files of the selected user-defined metrics in batches.
Batch publish	You can publish the selected user-defined metrics in batches and deliver them to related cluster nodes.

Operation	Description
Batch offline	You can bring selected user-defined metrics offline in batches and delete information about user-defined metrics and scheduled tasks from related cluster nodes.
Batch delete	You can delete the selected user-defined metrics, information about user-defined metrics and scheduled tasks on related cluster nodes, and user-defined metric records.

----End

Performing Operations on a User Defined Monitoring Metric

Step 1 Log in to ServiceCM.

Step 2 In the navigation pane, choose **Monitoring Alarms** > **Define Monitoring**. You can enable, disable, publish, take offline, delete, and view details about a user-defined metric. The procedure is as follows:

Table 4-3 Performing operations on a user defined monitoring metric

Navigation Path	Description
Stop	Delivers the command for stopping data collection for this metric to the related cluster node.
Enable	Delivers the command for enabling data collection for this metric to related cluster nodes.
Publish	Delivers this metric to related cluster nodes.
More > Offline	Brings offline information about user-defined metrics and scheduled tasks on related cluster nodes.
More > Delete	Deletes information about user-defined metrics and scheduled tasks on related cluster nodes, as well as user-defined metric records.
More >Details	Accesses the metric details page, on which you can modify the metric information and save the modification.

----End

Querying User-Defined Monitoring Metrics

Step 1 Log in to ServiceCM.

Step 2 In the navigation pane, choose **Monitoring Alarms** > **Define Monitoring**.

Step 3 In the search box on the right, you can search for a specified metric by plug-in name.

Step 4 You can click the setting icon in the search box to set the displayed columns. The metric plug-in and release status details are as follows:

Table 4-4 User-defined metric plug-in statuses

Plug-in Status	Operation	Description
Enabled	[Enable]	Enables data collection on this metric.
Stopped	[Stop]	Stops data collection on this metric.

Table 4-5 Publish statuses of user-defined metrics

Publish Status	Action	Description
Unpublished	[Batch import], [Add plugin]	Metric configuration data exists only in the DMS database and is not delivered to any node in the cluster. It is same as **Offline**. The opposite status is **Published**.
Published	[Batch publish], [Publish]	Deliver metrics to cluster nodes. The opposite statuses are **Offline** or **Unpublished**.
Offline	[Batch Offline], [Offline]	Metric configuration data exists only in the DMS database and is not delivered to any node in the cluster. It is same as **Unpublished**. The opposite status is **Published**.
Delete	[Batch Delete], [Delete]	The metric information and related scheduled tasks are deleted from related cluster nodes, the metric collection table is deleted from the DMS database, and the metric record is deleted from the metric table.

----End

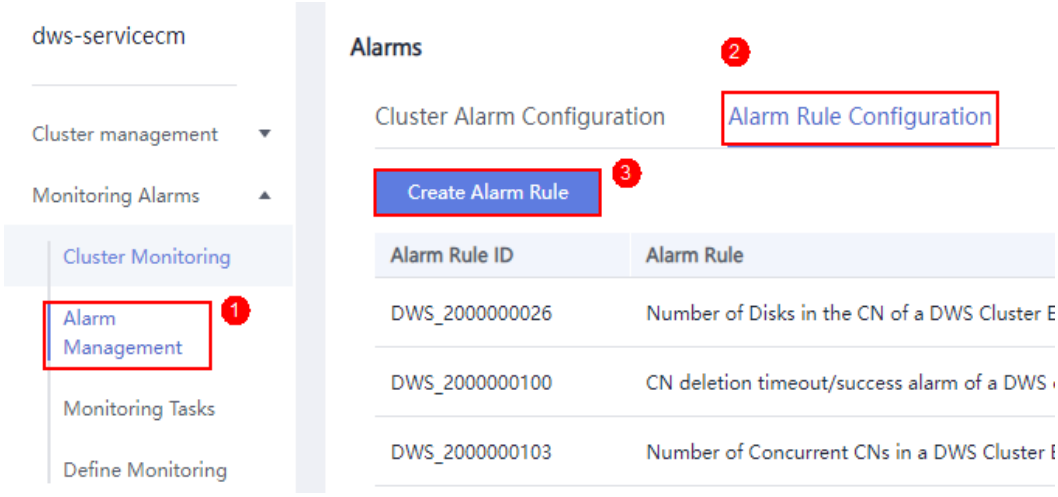
5 User-defined Alarm Thresholds

5.1 Alarm Settings

Adding Customized Monitoring Alarms

Step 1 Log in to ServiceCM.

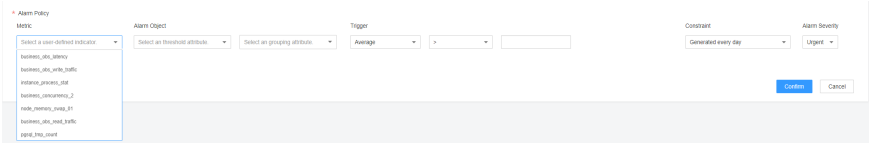
Step 2 In the navigation pane on the left, choose **Monitoring Alarms** > **Alarm Management**. On the displayed page, click **Alarm Rule Configuration**. Click **Create Alarm Rule**. The **Create Alarm Rule** page is displayed.



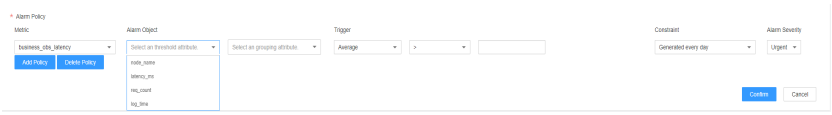
Step 3 You can configure items, such as the alarm rule name, rule description, clusters associated with the rule, and alarm policy.

- **Alarm Rule Type**: Select **Metric Alarm**.
- **Alarm Rule**
- **Description**
- **Alarm Validity Range**
- **Associated Cluster**: From the drop-down list, select the current tenant's clusters to which the alarm rule applies.

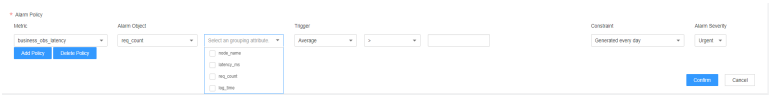
- **Triggered Policies**
  - **Independent**: Alarm policies are triggered independently of each other.
  - **Priority**: Alarm policies are triggered by priority. Policies of a lower priority will be automatically triggered after those of a higher priority.
- **Alarm Policy**
  - **Metric**: Select a user-defined metric from the drop-down list. For details about user-defined metrics, see **User-defined Monitoring Metrics**.

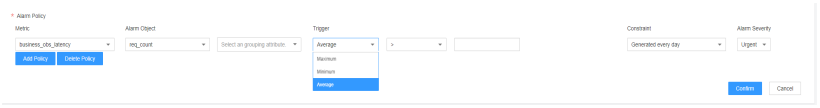
  - **Alarm Object**: Select a threshold attribute from the first drop-down list. The threshold attribute is the database field to be compared with the threshold. Only the numeric type is supported.

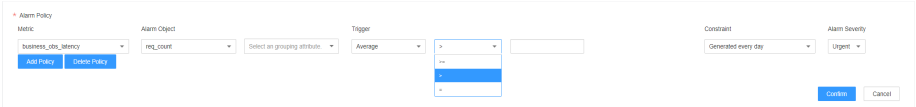
    

    Select a grouping attribute from the second drop-down list box. The data reported by monitoring metrics may contain information about different nodes, instances, users, and databases. Therefore, you need to group the data when calculating the extremum value and average value to avoid confusion of data in different dimensions and improve the alarm accuracy. The group attribute is optional. You are advised to select a non-numeric field.

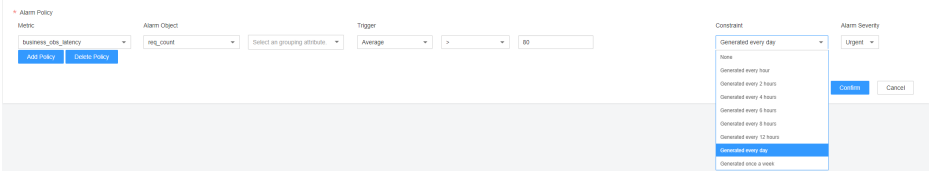
    

  - **Trigger**: defines the calculation rule for threshold-based alarm determination. It includes the calculation method, comparison method, and threshold of the selected data. Select the average value within a period of time of a metric to reduce the probability of alarm oscillation.

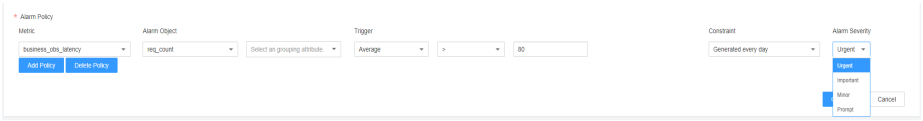
    

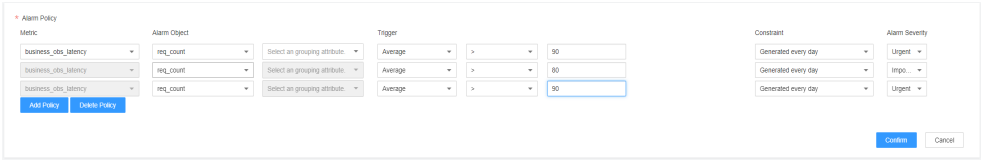
  - **Constraint**: suppresses the repeated triggering and clearance of alarms of the same type within the specified period.

  - **Alarm Severity**: includes **Urgent**, **Important**, **Minor**, and **Prompt**.



Step 4 The alarm policy allows you to configure alarms for different database columns and configure different thresholds and alarm severities for the same database column. You can click **Add Policy** or **Delete Policy** to add or delete an alarm policy.



----End

Managing User-defined Alarms

You can modify, disable, enable, and delete user-defined monitoring alarms. The procedure is as follows:

Step 1 Log in to ServiceCM.

Step 2 In the navigation pane on the left, choose **Monitoring Alarms** > **Alarm Management**. On the displayed page, click **Alarm Rule Configuration**.

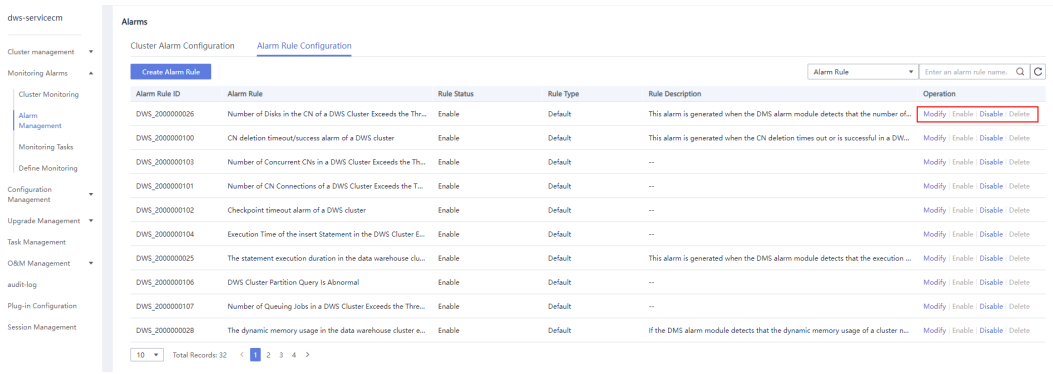


Table 5-1 Managing User-defined Alarms

Operation	Description
Stop	Disables the custom monitoring alarm metric.
Enable	Enables the custom monitoring alarm metric.
Modify	Modifies policies of the custom monitoring alarm metric.
Delete	Deletes the custom monitoring alarm metric.

----End

