Database and Application Migration UGO 24.9.0

User Guide

Issue	01
Date	2024-09-30

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 Database Evaluation

UGO collects basic information and performance data of a source database, as well as SQL statements of specific object types, and provides an overview of the source database. Based on the analysis of factors such as compatibility and object complexity, UGO provides an analysis report on the compatibility and reconstruction cost of the source and target databases, helping you select a suitable target database and evaluate migration workloads.

1.1 Viewing Data Collection Logs

Scenarios

Data collection logs record the events that occur in a project and the time when the events occur.

Procedure

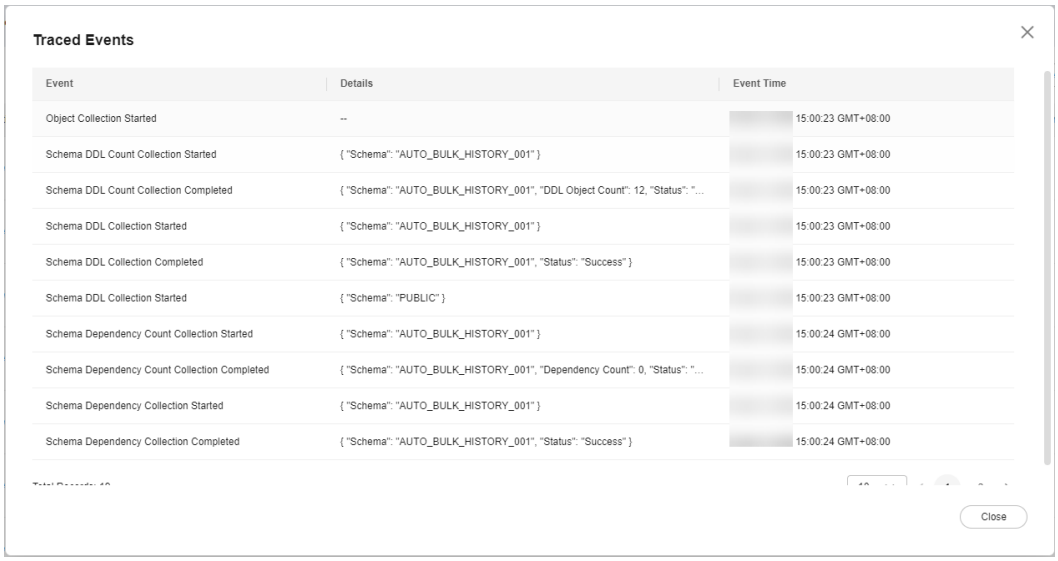
Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Schema Migration** > **DB Evaluation**.

Step 3 In the evaluation project task list, select the project to be analyzed and click **Trace** in the **Operation** column. The collection trace list is displayed.

The detailed information includes the schema names, number of collected logs, and log collection statuses of events.

Figure 1-1 Collection log details



- **Schema**: schema name.
- **DDL Object Count**: Number of collection objects in the schema.
- **Status**: log collection status.

----End

1.2 Re-Collecting Objects

Prerequisites

- The **Connection Type** of a project is **Online**.
- You have required permissions of a project.
- An evaluation project has been created but the target database has not been confirmed.
- There are objects that are not collected in the source database of a project.
- Frozen accounts cannot re-collect objects.

Procedure

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Schema Migration** > **DB Evaluation**.

Step 3 In the evaluation project list, click the name of an online project whose **Project Status** is **In Progress-Object Collection Error**. The **Source DB Analysis** page is displayed.

Step 4 Click **View Object Details** on the right of the **Object Statistics** area. The database object list page is displayed.

Step 5 Select objects that are not collected and click **Re-collect** above the list.

Step 6 In the displayed dialog box, click **OK**. A message is displayed, indicating that the re-collection task is submitted.

The re-collection is performed asynchronously. You need to wait until the re-collection is complete and refresh the page.

----End

1.3 Re-Evaluate a Project

Prerequisites

The **Project Status** of an evaluation project is **In progress. Confirm Target DB Pending**.

Frozen accounts cannot re-evaluate objects.

NOTE

This function is not available for Microsoft SQL Server databases.

Scenarios

You can re-evaluate the information of your project, including dynamic SQL statements, workloads, object conversion, top incompatible syntax points, and system objects.

Procedure

Step 1 Log in to the UGO console.

Step 2 In the navigation pane, choose **Schema Migration** > **DB Evaluation**.

Step 3 In the evaluation project task list, select the project to be analyzed and choose **More** > **Re-Evaluate** in the **Operation** column.

Step 4 (If the source database type is Oracle) In the displayed box, click **Edit** and configure **Dynamic SQL Evaluation**.

- **Yes**: The dynamic SQL statements in objects are analyzed.
- **No**: The dynamic SQL statements are not analyzed.

Step 5 Click **OK**.

- **Project Status** of the evaluation project is **In progress**. The detailed project status and progress are also displayed.
- To stop or resume the re-evaluation, click **Stop** or **Resume**.

----End

1.4 Performing an Incremental Evaluation

NOTE

- This function is available only for the source databases Oracle and MySQL.
- If the source database is MySQL, the incremental USER, GRANT, and ROLE objects cannot be collected.

Prerequisites

- You have required permissions of a project.
- An evaluation project has been created but the target database has not been confirmed.
- Tables, functions and other data are added to the source database in a project. Deleted or modified data cannot be identified.
- Before performing incremental evaluation, you must perform differential analysis.
- Frozen accounts cannot perform an incremental evaluation.

Procedure

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Schema Migration** > **DB Evaluation**.

Step 3 In the evaluation project task list, select the project to be analyzed and choose **More** > **Run Differential Analysis** in the **Operation** column.

Step 4 After the differentiation analysis is complete, choose **More** > **Perform Incremental Evaluation** in the **Operation** column

Step 5 After the incremental evaluation is complete, view the updated project information. For details about the project, see **Viewing the Database Evaluation Result**.

----End

1.5 Resuming a Stopped Project

Prerequisites

- **Project Status** of an evaluation project is **Stopped**.
- Frozen accounts cannot resume a stopped project.

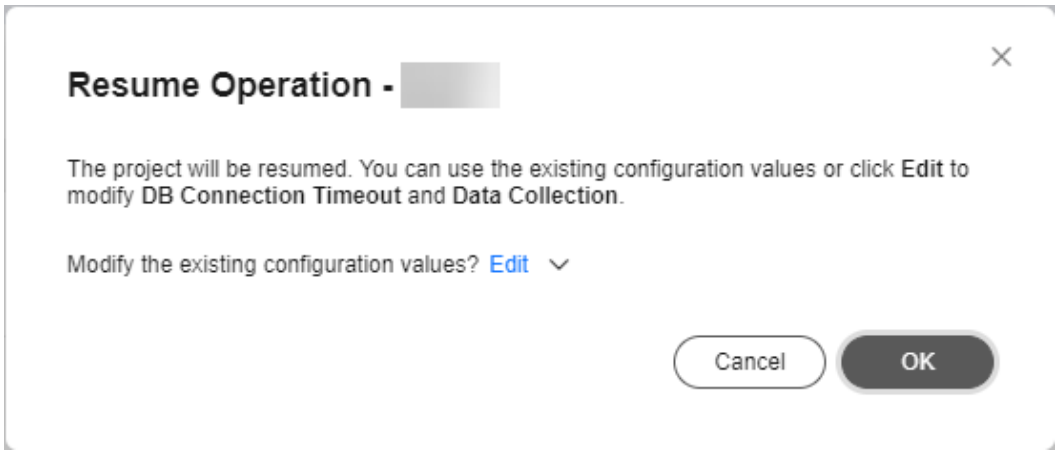
Procedure

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Schema Migration** > **DB Evaluation**.

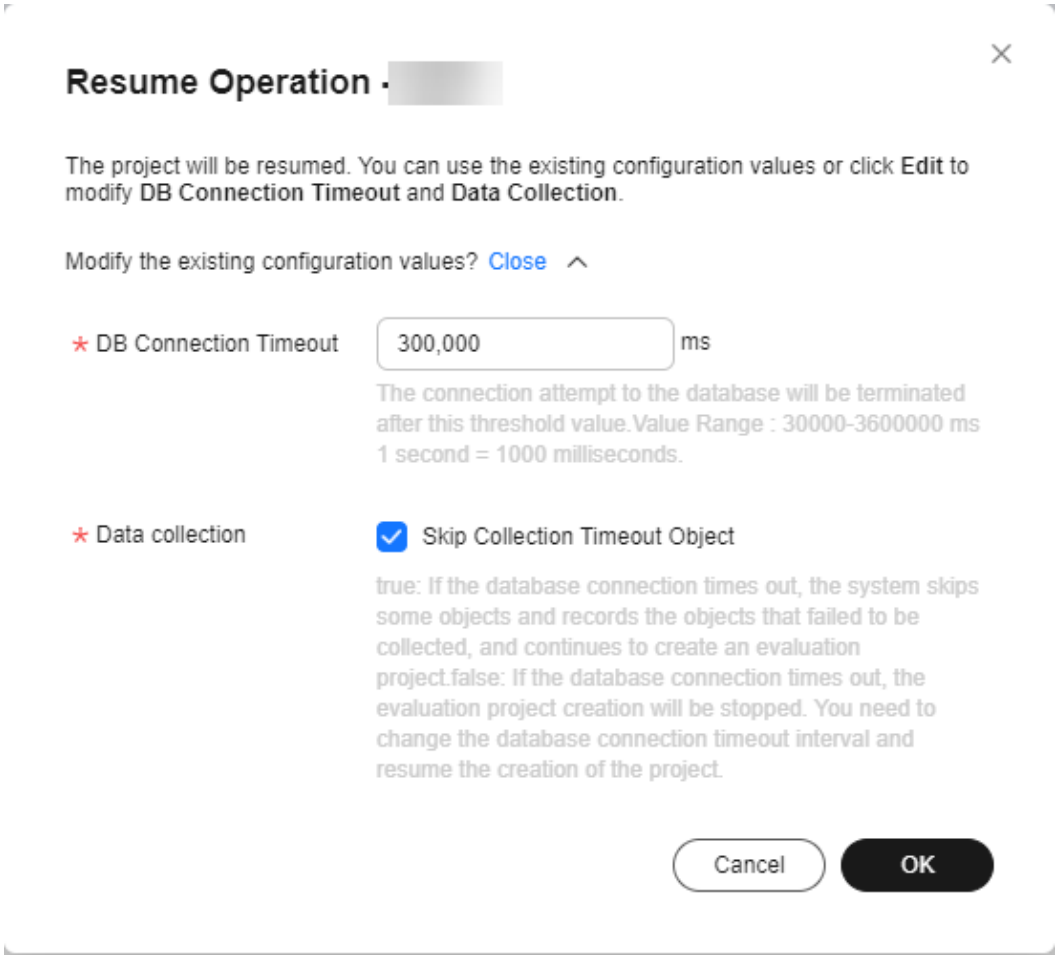
Step 3 In the evaluation project list, locate the project in the **Stopped** status and click **Resume** in the **Operation** column.

Figure 1-2 Resuming an evaluation project



Step 4 To prevent database collection from occupying resources for a long time, you can configure the collection timeout period. To modify **DB Connection Timeout**, click **Edit**. Then, select **Skip Collection Timeout Object** for **Data collection**.

Figure 1-3 Editing settings



- **DB Connection Timeout**: time limit within which the system attempts to connect to the database, in millisecond. The value ranges from **30000** to **3600000**.

- **Skip Collection Timeout Object**
  - Select this option: If the database connection times out, the system skips some objects and records the objects that failed to be collected, and continues to create an evaluation project.
  - Deselect this option: If the database connection times out, the evaluation project creation will be stopped. You need to change the database connection timeout and resume the creation of the project.

Step 5 To apply the current settings, click **OK**.

----End

1.6 Running Differentiation Analysis and Viewing the Generated Report

NOTE

This function is not available for Microsoft SQL Server and PostgreSQL databases.

Prerequisites

- The **Project Status** of a project is **Completed. Create Migration Project**.
- Frozen accounts cannot run differentiation analysis.

Scenarios

You can run a difference analysis of collected data during database evaluation.

Running differentiation analysis is to learn about the differences between the recent evaluation and the previous evaluation. The added or deleted objects can be displayed in the differentiation analysis, but the modified objects cannot be displayed.

If collected data does not change, the difference analysis report remains unchanged when you run the analysis.

Procedure

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Schema Migration** > **DB Evaluation**.

Step 3 In the evaluation project task list, select the project to be analyzed and choose **More** > **Run Differential Analysis** in the **Operation** column.

A message **Differential Analysis Scheduled** is displayed.

Step 4 After the **Project Status** shows **Differential Analysis Generated**, choose **More** > **View Differential Analysis Report** in the **Operation** column.

Step 5 In the **Differential Analysis Report** page, view the schema names, object types, object names, and supported operations.

----End

1.7 Managing a Tag

Scenarios

Tag Management Service (TMS) enables you to use tags on the management console to manage resources. TMS works with other cloud services to manage tags. TMS manages tags globally, and other cloud services manage their own tags.

- You are advised to configure predefined tags on the TMS console.
- A tag consists of a key and value. You can add only one value for each key.
- Each DB instance can have up to 10 tags.
- You can add tags when creating an evaluation project or a migration project, or add tags after the project is created.

Adding a Tag

Step 1 Log in to the UGO console.

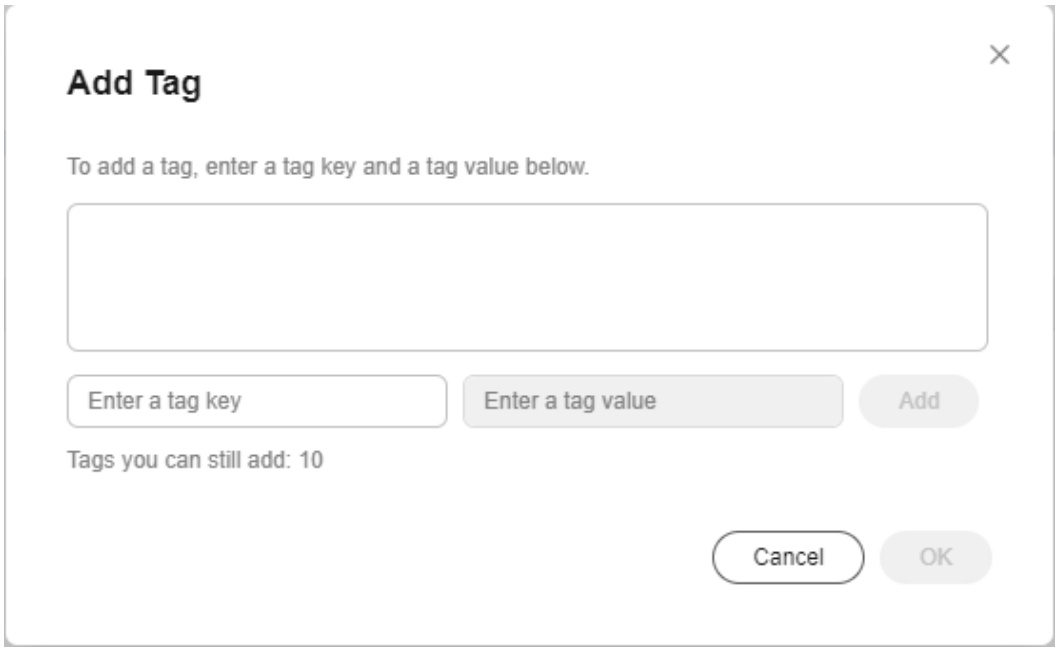
Step 2 In the navigation pane on the left, choose **Schema Migration** > **DB Evaluation** or **Schema Migration** > **Object Migration**.

Step 3 Click the project name.

Step 4 On the **Tags** page, click **Add Tag**. In the displayed dialog box, enter a tag key and value, and click **Add**.

Step 5 After all tags are added, click **OK** in the lower part of the dialog box. You can view and manage your tags of the current page.

Figure 1-4 Adding a tag



- When you enter a tag key and value, the system automatically displays all predefined tags associated with the tag.
- The tag key cannot be empty and must be unique. It can contain up to 128 characters, including letters, digits, spaces, and special characters (_.:=+-@), but cannot start with _**sys**_ or a space and end with a space.
- A tag value can contain up to 255 characters, including letters, digits, spaces, and special characters (_.:/=+-@), but cannot start and end with a space.

----End

Editing a Tag

Step 1 Log in to the UGO console.

Step 2 In the navigation pane on the left, choose **Schema Migration** > **DB Evaluation** or **Schema Migration** > **Object Migration**.

Step 3 Click the project name.

Step 4 Locate the tag to be edited and click **Edit** in the **Operation** column. In the displayed dialog box, change the tag value and click **OK**.

- Only the tag value can be edited.
- A tag value can contain up to 43 characters, cannot start and end with a space, and cannot contain the special characters (=*<>,|/).

Step 5 After the tag is edited, view and manage it in the current page.

----End

Deleting a Tag

Step 1 Log in to the UGO console.

Step 2 In the navigation pane on the left, choose **Schema Migration** > **DB Evaluation** or **Schema Migration** > **Object Migration**.

Step 3 Click the project name.

Step 4 Locate the tag to be deleted and click **Delete** in the **Operation** column. In the displayed dialog box, click **OK**.

Step 5 After the tag is deleted, view that there is no the tag in the current page.

----End

NOTE

Tag information cannot be deleted by accounts frozen for security reasons. All frozen accounts cannot modify and add tags.

1.8 Deleting an Evaluation Project

- Deleted projects cannot be recovered.
- Evaluation projects frozen for security reasons cannot be deleted.

Prerequisites

- The **Project Status** of an evaluation project is **Completed. Create Migration Project** or **In progress. Confirm Target DB Pending**.
- The evaluation project is not associated with any migration project.

Procedure

Step 1 Log in to the UGO console.

Step 2 In the navigation pane, choose **Schema Migration** > **DB Evaluation**.

Step 3 In the evaluation project list, locate the project you want to delete and click **Delete**.

Step 4 In the displayed dialog box, click **OK**.

----End

2 Object Migration

After the database evaluation is complete and the target database is confirmed, you can start to migrate database objects with a few clicks. In this module, UGO guides you to make a conversion plan and design a solution to automatically convert syntax. For objects that fail to be converted or migrated, you can edit and batch modify them to simplify manual reconstruction. Each modification is recorded, so that you can view and roll back historical modifications.

2.1 Object Migration Implementation

2.1.1 Making a Conversion Plan

Scenarios

After a migration project is created, select objects to be migrated on the conversion plan page. You can select an object or a type of objects in batches.

Constraints

- If the source database type is GoldenDB or MySQL (earlier than 8.0), there is no ROLE object in the source database. UGO does not collect ROLE objects and the number of ROLE objects is displayed as **0**.
- To migrate data, you are advised to skip the trigger migration temporarily. Otherwise, triggers may change data during migration. After data is migrated, migrate the triggers.
- If **Object Status** of objects is **Abnormal**, their **Conversion Status** is **Skip**. It means that abnormal objects cannot be converted.

Prerequisites

- The project status is **Ready**, the target database information is correct, and the connection test is successful.
- The user connected to the target database must have the permissions to create, delete, and modify objects such as schemas, tables, programs, indexes, users, functions, and views. For details, see **Viewing a Permission Check Report**.

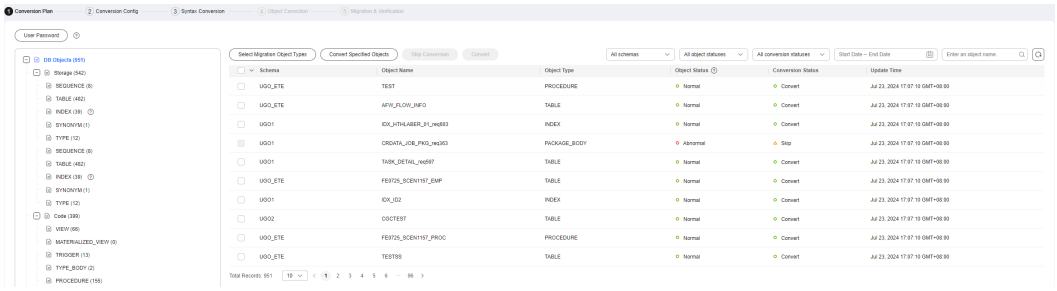
Converting Specified Objects

Step 1 Log in to the UGO console.

Step 2 In the navigation pane, choose **Schema Migration** > **Object Migration**.

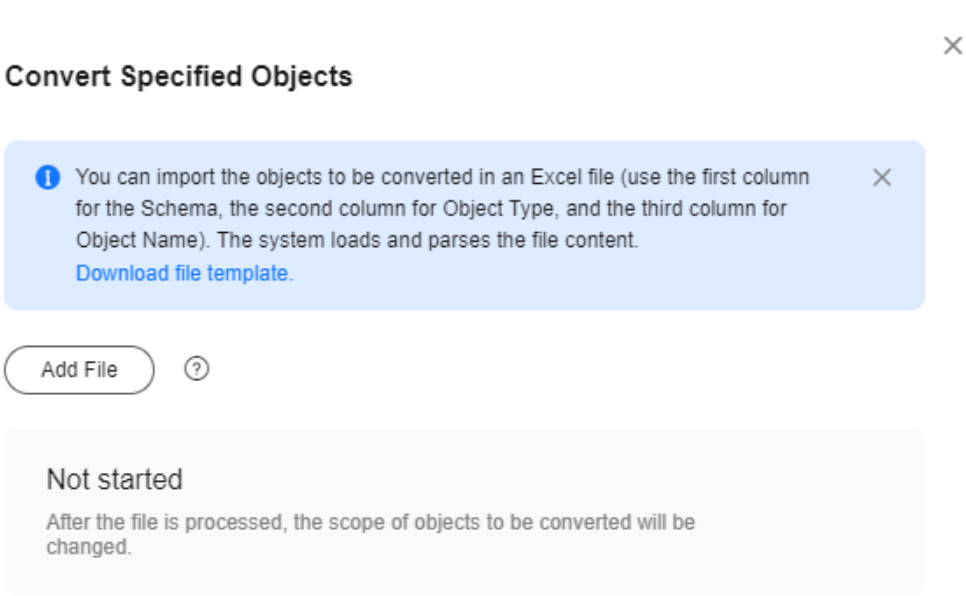
Step 3 Locate the project to be migrated and click **Migrate** in the **Operation** column.

Figure 2-1 Conversion plan



Step 4 Click **Convert Specified Objects**. The **Convert Specified Objects** dialog box is displayed.

Figure 2-2 Converting specified objects



- Click **Download file template** to download a template.
  - In the Excel file, the first column indicates schema, the second column indicates object type, and the third column indicates object name. The three columns specify an object and cannot be empty.
  - Click **Convert Specified Objects** to upload an Excel file. Identify all objects that meet the schema, object type, and object name requirements in the list with **Convert** displayed.
- Click **Add File** to upload a file where there are objects to be converted.

- The maximum size of the file is 30 MB, and the maximum number of rows is 65,535 (excluding the table header). If the content is incorrect, you can click **View Error** in the upload result to view the error cause. After the error is rectified, you can upload the file again.
- After the file is uploaded successfully, the conversion status of objects is automatically updated.

----End

Distribution Mapping of Table Objects

If the source database type is Oracle or MySQL and the target database type is GaussDB Distributed, you can perform the following operations:

- Replicating tables: Select one or more objects and click **Replicate Table** above the list. The object name is suffixed with "Replicate Table".
- Modifying distribution mapping information: Locate an object whose **Object Type** is **TABLE** and click **Distribution Mapping** in the **Object Type** column. Change the distribution type to **HASH** or **REPLICATION** and click **Apply**. The object name is suffixed with "replicate table" or "hash (*Column name*)".
- Deleting distribution mapping information: Browse to a table object and click **Distribution Mapping** in the **Object Type** column. In the displayed dialog box, select **Delete distributed mapping** and click **Apply**. **Distribution Mapping** next to the object name will not be displayed.

  NOTE

  If you select **Hash**, all fields that meet the data type constraints of GaussDB distribution columns are displayed.

2.1.2 Setting Conversion Configuration Items

CAUTION

Conversion configuration items directly affect the success rate of syntax conversion and migration result verification. Therefore, you need to correctly set these configuration items and select a suitable conversion plan.

Syntax Conversion Configuration

The conversion scenarios controlled by each feature parameter are different. It is difficult for users and service personnel to systematically and comprehensively understand the conversion. Multiple reworks affect migration efficiency and user experience. You can use conversion configuration to reduce the manual update configuration and quickly and accurately complete the migration. You also need to map tablespaces between the source and target database.

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Schema Migration** > **Object Migration**. The migration project list is displayed.

Step 3 Locate the project to be migrated and click **Migrate** in the **Operation** column. Click **Conversion Config**.

Step 4 Select a value from the **Baseline Template** drop-down list. Create a custom conversion configuration template. For details, see **Creating a Configuration Template**.

- Two templates are preset: default template and maximum compatibility template.
- After you select a template, the preset values in the template are imported to the configuration item list.

Step 5 Set the configuration items based on the actual conversion scenario.

- In the **Current Configuration** column, move the cursor to ⓘ of each feature to view the impact of the feature. You can click **View Sample** to view details about the configuration information and the current configuration conversion example.
- Locate a feature and click **Edit** in the **Operation** column to modify the current configuration of the feature and click **OK**.
- After you import configuration parameters, the configuration status of features is **Default value**. If you edit the configuration of a feature, the configuration status will become **Modified**.
- If the custom template in your project is deleted, the template configurations are still used, but **Parameter Configuration** is displayed as **Default value**.

----End

NOTE

The system view and advanced package are converted based on the latest GaussDB version. If the target database is of an earlier version, the SQL statements after conversion may be incompatible.

Mapping Tablespaces

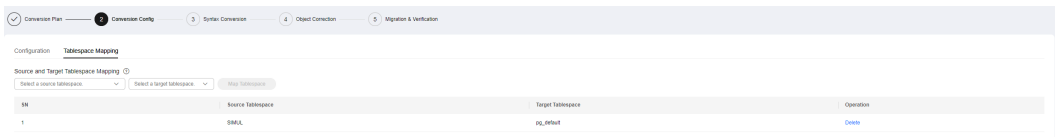
The source and target databases may have different tablespace names. Before migration, you need to map tablespaces between the source and target databases. Tablespace names of the source statement can be converted to the mapped tablespace names of the target database.

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Schema Migration** > **Object Migration**.

Step 3 Locate a project you want to migrate and click **Migrate** in the **Operation** column. On the **Conversion Config** page, click **Tablespace Mapping**.

Figure 2-3 Tablespace Mapping



Step 4 Select the source and target database tablespaces from the drop-down list boxes and click **Map Tablespace**. The tablespaces are mapped.

----End

NOTE

- The tablespace to be mapped has been created in the target database.
- The following migration flows support the tablespace mapping function.
  - From Oracle to GaussDB
  - From Oracle to PostgreSQL
- You can map one tablespace of the source database to one tablespace of the target database or map multiple tablespaces of the source database to one tablespace of the target database.

2.1.3 Converting Syntax

Procedure

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Schema Migration** > **Object Migration**. The migration project list is displayed.

Step 3 Locate the project to be migrated and click **Migrate** in the **Operation** column. Click the **Syntax Conversion** tab.

Step 4 Click **Start** to start the conversion.

Step 5 Before the conversion is complete, click **Pause** to pause the conversion. Click **Resume** to continue the conversion.

Step 6 After the conversion is complete, click **Download Reports**. Locate the required report and click **Download** to download the report to the local PC for analysis.

- **Conversion Error Report**: This report contains details about source database objects whose syntax could not be converted to syntax compatible with the target database.
- **Anonymized Conversion Error Report**: This report contains details about anonymous source database objects whose syntax could not be converted to syntax compatible with the target database.
- **Conversion Risk Report**: This report contains details about objects converted with risks based on configurations. However, there are function differences after the conversion.
- **Anonymized Conversion Risk Report**: This report contains details about anonymous objects converted with risks. However, there are function differences after the conversion.
- **Converted SQL Parsing Failure Report**: This report contains details about source database objects that could not be parsed using the conversion script for the target database syntax.
- **Anonymous Converted SQL Parsing Failure Report**: This report contains details about anonymous source database objects that could not be parsed using the conversion script for the target database syntax.

- Locate an object type that failed to be converted, and click **Details** in the **Operation** column to go to the **Object Correction** page to view details.

----End

Viewing Syntax Conversion History

Step 1 Log in to the UGO console.

Step 2 In the navigation pane, choose **Schema Migration** > **Object Migration**. The migration project list is displayed.

Step 3 Locate the project to be migrated and click **Migrate** in the **Operation** column. On the displayed page, click the **Syntax Conversion** tab.

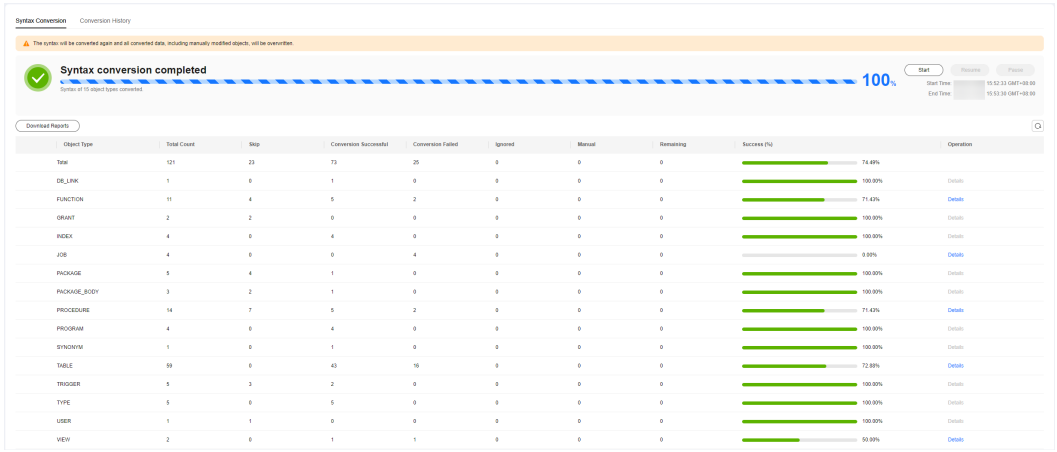
Step 4 Click **Conversion History**.

The conversion information is displayed in chronological order.

Figure 2-4 Conversion history

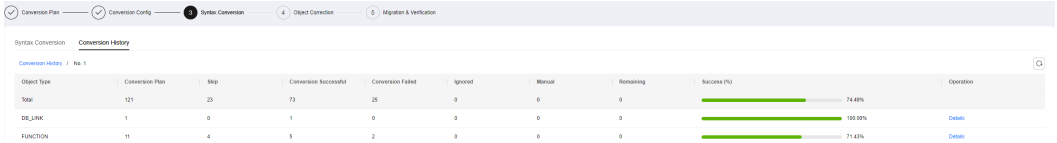
Step 5 Click **Details** to view information such as the object type, conversion plan, and conversion status (successful or failed).

Figure 2-5 Conversion details



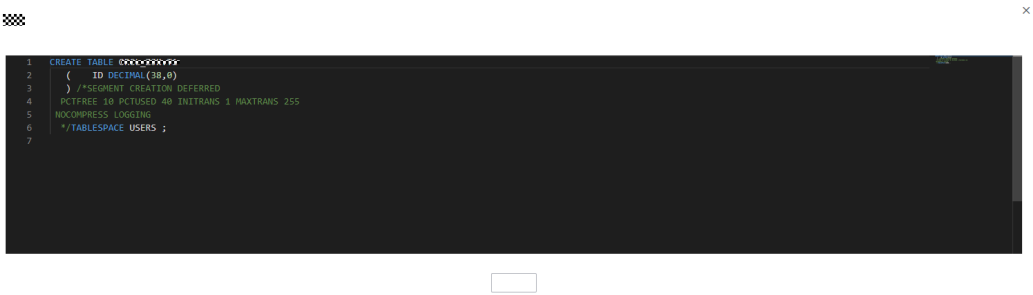
Step 6 Locate an object type and click **Details** to view its details, including the schema, object name, migration status, and conversion status. You can search for an object by name or view details about each schema.

Figure 2-6 Object type details



Step 7 Locate an object and click **Details** to view the code.

Figure 2-7 Object details



----End

2.1.4 Correcting Objects

WARNING

The modified statements will run on the target database. Do not use statements that cause the drop of other tables and object types, or statements that cause malicious activities such as consuming too much CPU, memory and other side effects over the target database.

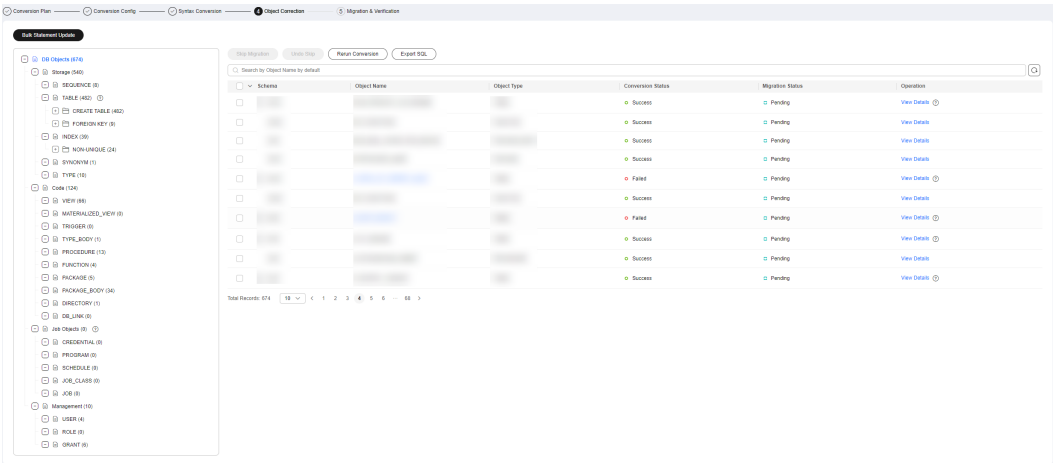
Correcting Objects That Failed to Be Converted

Step 1 Log in to the UGO console.

Step 2 In the navigation pane, choose **Schema Migration** > **Object Migration**. The migration project list is displayed.

Step 3 Locate the project to be migrated and click **Migrate** in the **Operation** column. On the displayed page, click the **Object Correction** tab.

Figure 2-8 Object correction



- You can view the schema name, object name, object type, conversion status, and migration status.
- You can select a filter (**Object Type**, **Schema**, **Conversion Status**, **Migration Status**, **Object Name**, **SQL Content Includes**, **SQL Content Excludes**, **Conversion Errors Include**, or **Migration Errors Include**) to search for objects. If you select **Object Name**, fuzzy search is supported. If you select **SQL Content Includes** or **SQL Content Excludes**, you can enter up to three keywords and each keyword contains up to 10 characters. If there are a large number of objects in a project, the query slows.
- If you search for the desired object by schema, the schema name cannot contain whitespace characters.
- Objects such as TABLE and INDEX may be split into multiple subobjects.
  - Splitting objects can control the migration sequence of multiple features of an object.
  - To view the status and content of all child objects, click **View Details** next to the parent object.
  - Splitting objects depends on the conversion function of UGO. If an error occurs during script parsing or conversion, objects are not split.
  - The statuses of subobjects and other objects are different.
    - If the conversion status of a parent object is **Success**, the conversion status its child object is also **Success**.
    - If the conversion status of a parent object is **Failed**, the conversion status of at least one child object is **Failed**.
    - If the conversion status of a parent object is **Manual**, the conversion status of at least one child object is **Manual**.
    - If the migration status of a parent object is **Success**, the migration status of its child objects is **Success** or **Ignore**.
    - If the migration status of a parent object is **Failed**, the migration status of at least one child object is **Failed**.
    - If the migration status of all child objects is **Ignore**, the migration status of their parent objects is also **Ignore**.

Step 4 Locate an object and click **View Details** in the **Operation** column.

NOTE

Correct management, storage, code, and job objects in sequence.

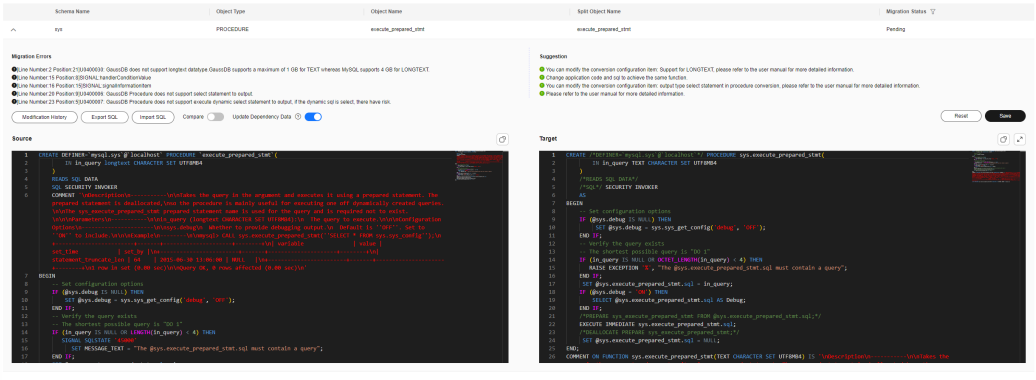
Step 5 View the conversion or migration error information. If any, modification suggestions are displayed.

- Locate an object whose **Conversion Status** is **Failed**, click **View Details** in the **Operation** column to view conversion errors.
- Locate an object whose **Migration Status** is **Failed**, click **View Details** in the **Operation** column to view migration errors.

NOTE

If the target database is GaussDB, database object names must meet start with a letter or underscore (_) and can contain letters, digits, underscores (_), dollar signs ($), and number signs (#). The length of a non-time series table cannot exceed 63 bytes, and the length of a time series table cannot exceed 53 characters.

Figure 2-9 Conversion errors on the object details page



Step 6 Modify SQL statements.

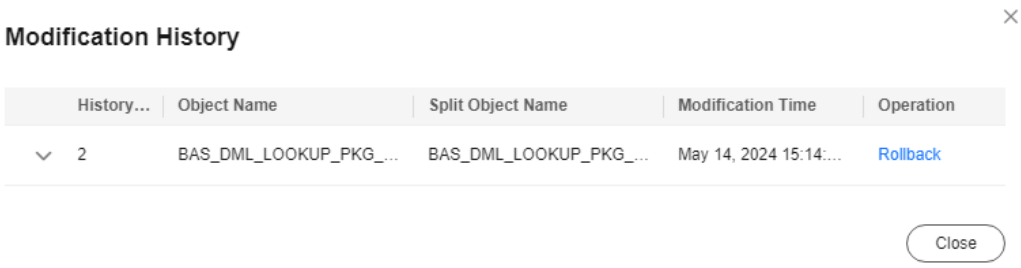
- Click **Export SQL** to export the SQL file of the target database. Exported file name: *Object name*.**sql**.
- Click **Import SQL** to directly import the modified SQL file. After the SQL file is uploaded, all changes are automatically saved.
- **Compare**: This switch is disabled by default. If this function is enabled, the detailed code comparison between the source and target databases is displayed. In this case, the SQL statements cannot be edited.
- Click next to **Source** or **Target** to copy the SQL statements to the clipboard.
- Click next to the target end to enter the full-screen editing mode. To exit the full-screen editing, click or click the edit box and press **Esc**.
- **Update Dependency Data**: If this switch is enabled, all dependent objects will also be verified. This function is enabled by default and can be disabled.
- Before saving the modified code, you can click **Reset** to clear all modifications. If the modification has been saved, you can only perform a rollback operation.
- Click **Save** to save the modifications for target database code. The modification records will be generated and saved on the **Modification History** page.

Step 7 After the modification is complete, click **Save** to generate a modification record. Click **Modification History** to view historical records.

In the **Modification History** area, locate the history record and click **Rollback** in the **Operation** column to roll back to the latest modification record.

Step 8 Click the drop-down icon next to **History ID** to compare the code before and after modification.

Figure 2-10 Modification history



----End

NOTE

- If you select Oracle as the source database type, DB_LINK objects can only be converted and cannot be migrated, and their migration status is **Ignore** by default.
- After the modification, the conversion status or migration status of the object changes from **Failed** to **Manual**.

Updating Statements in Batches

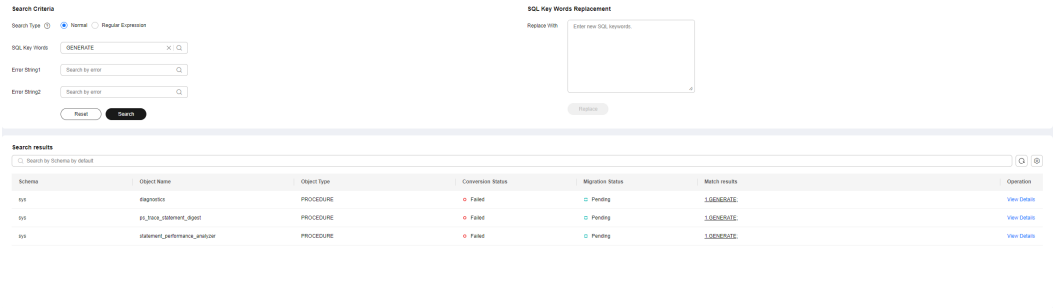
Step 1 Log in to the UGO console.

Step 2 In the navigation pane, choose **Schema Migration** > **Object Migration**. The migration project list is displayed.

Step 3 Locate the project to be migrated and click **Migrate** in the **Operation** column. On the displayed page, click the **Object Correction** tab.

Step 4 Click **Bulk Statement Update**.

Figure 2-11 Batch modification



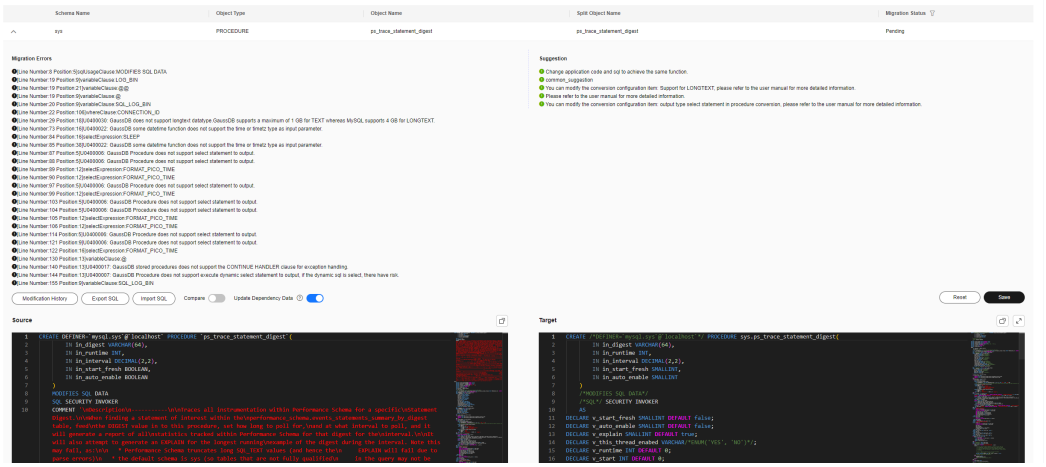
Step 5 On the **Search results** area, locate an object and click **View Details** in the **Operation** column.

You can view the schema name, object type, object name, split object name, conversion status, and migration status.

Step 6 View the conversion or migration error information. If any, modification suggestions are displayed.

- Locate an object whose **Conversion Status** is **Failed**, click **View Details** in the **Operation** column to view conversion errors.
- Locate an object whose **Migration Status** is **Failed**, click **View Details** in the **Operation** column to view migration errors.

Figure 2-12 Object details



Click the **Object Correction** tab. The SQL statements of the source and target databases are displayed.

Click next to **Source** or **Target** to copy the SQL statements to the clipboard.

Step 7 Copy a SQL keyword or character string, return to the batch modification page, and configure parameters as needed/

Figure 2-13 Batch modification

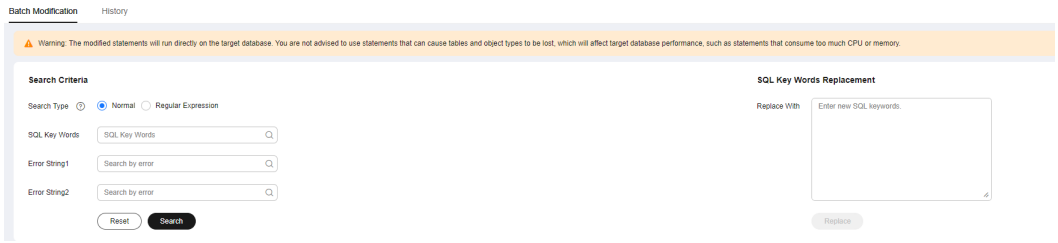


Table 2-1 Parameter description

Parameter	Mandatory	Description
Search Type	Yes	• **Normal**: simple search text type • **Regular Expression**: regular expression search text type **NOTE** – Incorrect use of regular expressions may cause unpredictable changes to SQL statements. Therefore, exercise caution when using regular expressions. – For details about regular expression specifications, see **Java Regex**.

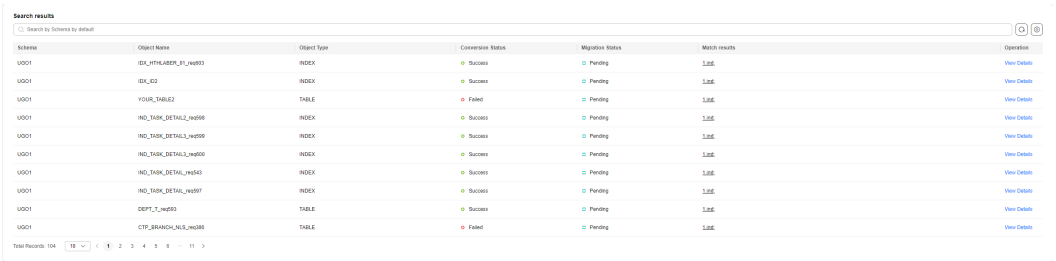
Parameter	Mandatory	Description
SQL Key Words	Yes	Search for keywords obtained in the **Object Details** page. Click **View SQL** in the **Operation** column to go to the **Object Details** page. For example, if you select **Regular Expression** and enter **DEFAULT*** as a keyword, the system searches for all target statements with the structure **DEFAULT** *Any SQL statement*.
Error String 1	No	Search for migration error messages obtained in the **Object Details** page. Click **View SQL** in the **Operation** column to go to the **Object Details** page.
Error String 2	No	Further search for migration error messages obtained in the **Object Details** page. Click **View SQL** in the **Operation** column to go to the **Object Details** page.

NOTE

- After the parameters are configured, click **Search**. Objects that meet search criteria are displayed in the lower part of the page.
- All error strings can be viewed in **Migration Errors** or **Conversion Errors** area, as shown in **Figure 2-12**.
- If the user connected to the target database is different from the schema owner, UGO will change the schema owner name to the username.

Step 8 Click **Search**. The objects that meet the search criteria are displayed.

Figure 2-14 Search results



You can search for objects by schema, object type, conversion status, and migration status.

Click the content in the **Match Results** column to view detailed matching information.

Step 9 Enter the modified SQL statement in the **Replace With** text box and click **Replace**.

Step 10 After the replacement is complete, click **Bulk Statement Update** and then click the **History** tab. On the displayed page, view the batch replacement records.

Figure 2-15 History

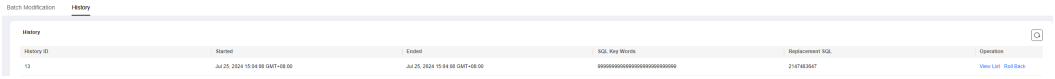
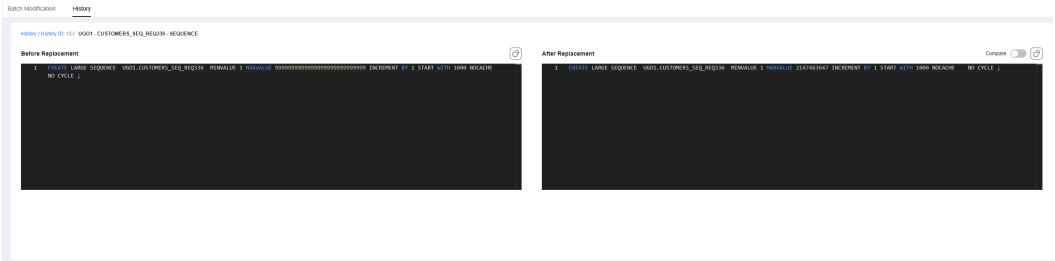


Table 2-2 Parameter description

Parameter	Description
History ID	The number of modification times is displayed in descending order.
Started	Time when the modification started, for example: 2021/09/15 16:28:15 GMT+08:00
Ended	Time when the modification ended, for example: 2021/09/15 16:28:16 GMT+08:00
SQL Key Words	SQL keywords to be searched for
Replacement SQL	Scripts or keywords to replace the original SQL statements
Operation	• **Roll Back**: You can roll back the last modification. • **View List**: You can view the history ID, schema, object name, and object type. Click **View SQL** to view the modified SQL statement.

Step 11 Click **View List**. On the displayed **History** page, click **View SQL** in the **Operation** column to view the SQL statements before and after the replacement.

Figure 2-16 History



----End

2.1.5 Verifying the Migration Result

Prerequisites

There is a project for which migration has been completed.

Procedure

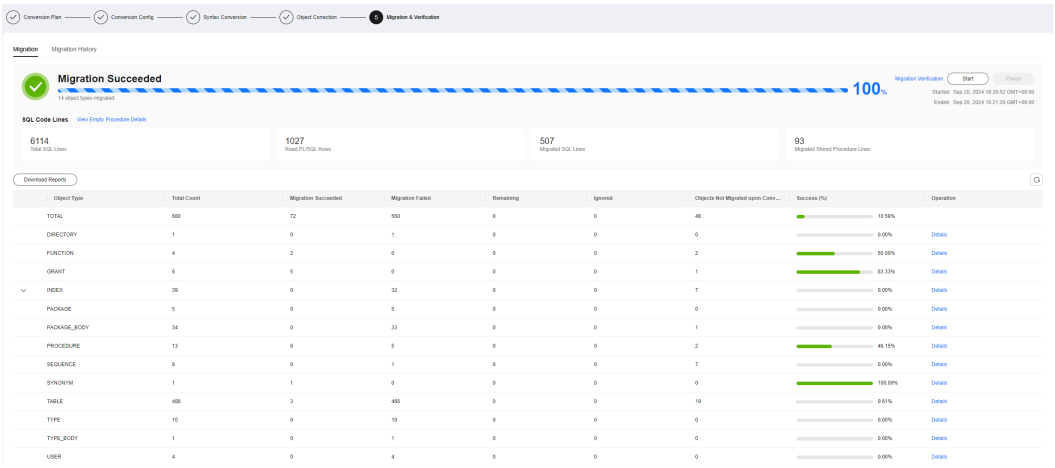
Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Schema Migration** > **Object Migration**. The migration project list is displayed.

Step 3 Locate the project to be migrated and click **Migrate** in the **Operation** column. On the displayed page, click the **Migration & Verification** tab.

The **Migration** tab page is displayed by default.

Figure 2-17 Going to the migration tab page

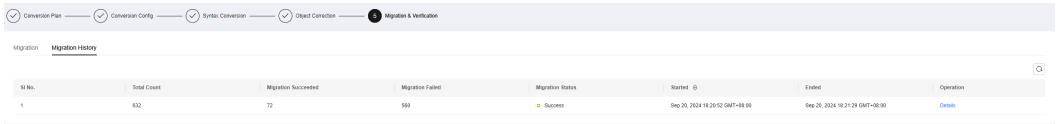


- Click **Migration Verification** to set the migration process. This parameter can be set only when the migration task is in the **Pending**, **The migration is being paused**, or **Migration Succeeded** state.
- The detailed information about the migration project is displayed, including object types, total objects, objects that have been migrated, objects that failed to be migrated, and objects that have not been migrated. If objects that failed to be converted will not be migrated, **Objects Not Migrated upon Conversion Failure** is also displayed.
- If the migration fails, click **Details** in the **Operation** column to go to the **Object Correction** page and view migration details of the object type.
- **View Empty Stored Procedure**: If the source database type is Oracle, you can view objects that fail to be created and failure occurrences.
- Click **Download Reports**, locate the required report, and click **Download** to download the report to the local PC for analysis. Data related to sub-objects is not included in the migration report.
  - **Migration statistical report**: This report includes a summary of object statuses during migration and verification.
  - **Migration error report**: This report includes failure details, such as statuses, migrated statements, and error details for each object.
  - **Anonymized error report**: This report consists of failure details, such as statuses, migrated statements, and error details for each object, but the original and migrated SQL statements will be anonymized.
  - **Full migration report**: This report includes failure details, such as statuses, migrated statements, and error details for each object.

Step 4 Click **Migration History**.

Detailed information of the migrated projects is displayed in the descending order. The information includes the serial number, total count, migration succeed, migration failed.

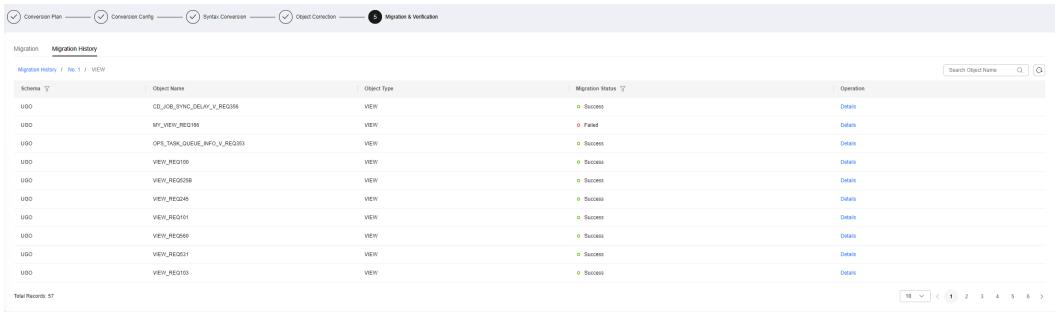
Figure 2-18 Migration history



Step 5 Click **Details** to view information such as the total count, migration succeed, and migration failed.

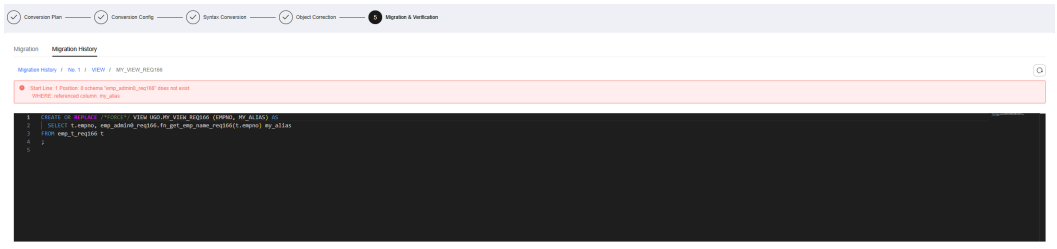
Step 6 Locate an object type and click **Details** to view its details. The detailed information includes the schema, object name, object types, and migration status. You can search for an object by name or view details about each schema.

Figure 2-19 Object type details



Step 7 Locate an object and click **Details** to view the code.

Figure 2-20 Object details



----End

2.2 Object Migration Project Management

2.2.1 Checking Permissions

Scenarios

Before data structure migration, check user permissions and database configurations to prevent failures caused by insufficient permissions.

Constraints

This function is not supported when the destination database type is GaussDB.

Prerequisites

After a migration project is created, if the permission check fails, **Project Status** is **Not ready**.

Procedure

Step 1 Log in to the UGO console.

Step 2 In the navigation pane, choose **Schema Migration** > **Object Migration**. The migration project list is displayed.

Step 3 Locate a project and click **More** > **Check Permissions** in the **Operation** column. A message is displayed, indicating that the permission check of the target database is started successfully.

- If the check is successful, **Project Status** is **Ready**.
- If the check fails, view the permission check report by following **Viewing a Permission Check Report**, modify the failed permissions in the target database, and check the permissions again.

----End

NOTE

If the target database is not GaussDB, during the permission check, UGO will create a stored procedure in the target database for permission check. After the check is complete, the stored procedure will be automatically deleted.

2.2.2 Viewing a Permission Check Report

Scenarios

You can view the permissions and status of a user used to connect to the target database in the migration project.

Prerequisites

- When creating a migration project, you can deselect **Skip Permission Check** and verify that the permission check is complete.
- Frozen accounts cannot be used to migrate objects or perform subsequent operations.
- The target database is not GaussDB.

Procedure

Step 1 Log in to the UGO console.

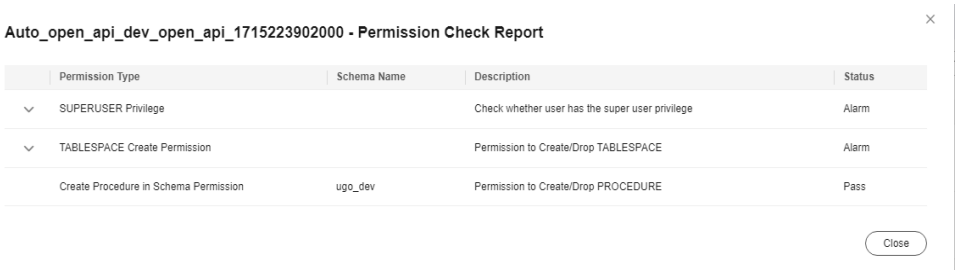
Step 2 In the navigation pane, choose **Schema Migration** > **Object Migration**. The migration project list is displayed.

Step 3 Locate a project and click **View Permission Check Report** in the **Operation** column.

The list displays the permission type, schema name, description, and status.

The permission status can be **Pass**, **Fail**, or **Alarm**. If there is a permission whose **Status** is **Fail**, the project status is **Not ready**.

Figure 2-21 Permission check list



Step 4 If there is a permission whose **Status** is **Alarm**, click to view its failure cause, details, solution, and hint.

----End

2.2.3 Deleting an Object Migration Project

Scenarios

There is an upper limit on the number of migration projects that can be created. You can delete completed migration projects to release resources.

Constraints

Deleted projects cannot be recovered.

Prerequisites

- A migration project has been created.
- Migration projects frozen for security reasons cannot be deleted.

Procedure

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Schema Migration** > **Object Migration**.

Step 3 In the migration project list, locate the project you want to delete and click **Delete**.

Step 4 In the displayed dialog box, click **OK**.

----End

2.2.4 Viewing Object Migration Details

To create a migration project, see **Migration Project**.

Scenarios

This section describes how to view details of created migration objects and the estimated migration success rate so that you can determine whether to implement the migration.

Procedure

Step 1 Log in to the UGO console.

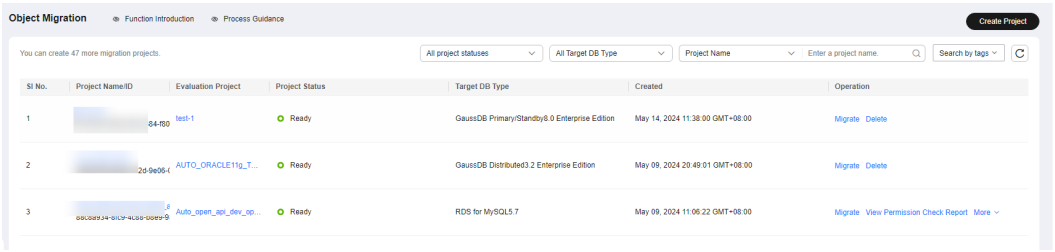
Step 2 In the navigation pane, choose **Schema Migration** > **Object Migration**.

If there is no migration project, the function introduction and process guidance are displayed by default. If there are migration projects, the function introduction and process guidance are hidden by default. You can expand them.

The following information is displayed in the migration project list: project name, project ID, evaluation project name, project status, target database type, creation date, and operation. You can directly copy the project ID.

If there are many projects, you can search for the project by database type, tag, project name, or ID.

Figure 2-22 Object migration



Step 3 Click the project name in the **Evaluation Project** column to view the detailed evaluation information of the project. For details, see **Viewing the Database Evaluation Result**.

Step 4 Click the project name in the **Project Name** column to view the information of the migration project.

Figure 2-23 Migration object details

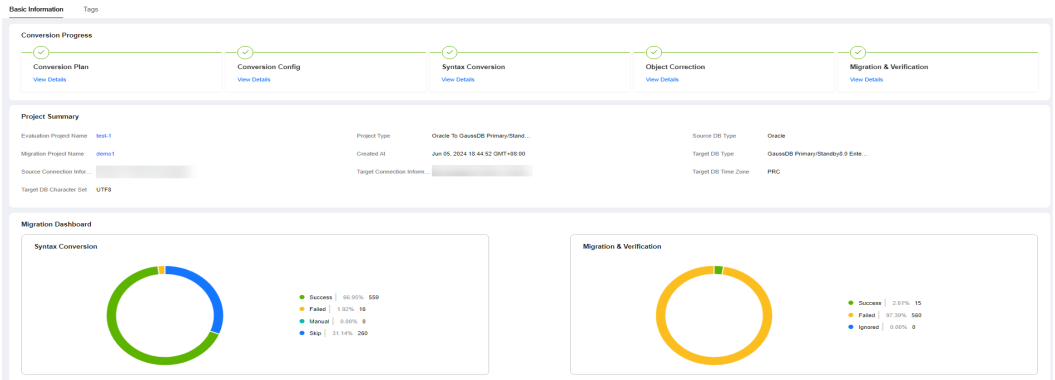


Table 2-3 Parameter description

Name	Description
Conversion Progress	Displays the project conversion progress. It consists of five steps: conversion plan, conversion configuration, syntax conversion, object correction, and migration & verification. If a step is complete, you can click **View Details** under the step to view its details. If a step is not performed, you can perform this step.
Project Summary	Displays the evaluation project name, project type, source database type, and target database type.
Migration Dashboard	Displays doughnut charts, percentages, specific values, and status of the syntax conversion and migration & verification. The migration & verification includes four statuses: **Success**, **Failed**, **In progress**, and **Ignored**.

Step 5 Click the **Tags** tab to add or modify tags. For details, see **Managing a Tag**.

----End

2.3 Migration Pre-check

When creating a migration project, check the permissions and character sets of a target database to prevent failures caused by insufficient permissions and incompatible character sets. You can view the current check items and modify database settings based on the failed check items.

2.3.1 Checking the Compatible Mode

If the target database type is GaussDB, the compatible mode will be checked. Different source databases use different compatible modes. If the target database uses a compatible mode different from a recommended one, a risk is reported. **Table 2-4** lists the GaussDB compatible modes.

Table 2-4 Compatible modes

Source Database	GaussDB Compatible Mode
MySQL or GoldenDB	MySQL-compatible mode
Others	Oracle-compatible mode

NOTE

The MySQL-compatible mode includes centralized B-compatible, M-compatible, and distributed MySQL-compatible modes. The Oracle-compatible mode includes centralized A-compatible and distributed ORA-compatible modes.

2.3.2 Checking Character Sets

The system checks character sets of the source and target databases. This check result does not affect subsequent operations. **Table 2-5** and **Table 2-7** describe the character set compatibility.

NOTE

Character set compatibility result:

- **Alarm**: The character sets of the source and target databases are incompatible.
- **Success**: The character sets of the source and target databases are compatible.

Table 2-5 Character set compatibility between Oracle and GaussDB

Source Database	Target Database	Result
SQL_ASCII	SQL_ASCII	Success
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Alarm
	UTF8	Alarm
	UTF32	Alarm
	Other character sets	Alarm
US7ASCII	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Alarm
	UTF8	Alarm
	UTF32	Alarm
	Other character sets	Alarm
ISO-8859-1	SQL_ASCII	Alarm
	ISO-8859-1	Success

Source Database	Target Database	Result
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Alarm
	UTF8	Alarm
	UTF32	Alarm
	Other character sets	Alarm
LATIN1	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Success
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Alarm
	UTF8	Alarm
	UTF32	Alarm
	Other character sets	Alarm
GB2312	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Success
	GBK	Success
	GB18030	Success
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
ZHS16GBK	SQL_ASCII	Alarm
	ISO-8859-1	Alarm

Source Database	Target Database	Result
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Success
	GB18030	Success
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
GBK	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Success
	GB18030	Success
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
ZHS32GB18030	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Success
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
GB18030	SQL_ASCII	Alarm
	ISO-8859-1	Alarm

Source Database	Target Database	Result
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Success
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
AL16UTF16	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
UTF16	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
AL32UTF8	SQL_ASCII	Alarm
	ISO-8859-1	Alarm

Source Database	Target Database	Result
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
UTF8	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
UTF32	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
Other character sets	SQL_ASCII	Alarm
	ISO-8859-1	Alarm

Source Database	Target Database	Result
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Alarm
	UTF8	Alarm
	UTF32	Alarm
	Other character sets	Alarm

Table 2-6 Character set compatibility between MySQL and GaussDB

Source Database	Target Database	Result
ISO-8859-1/LATIN1	SQL_ASCII	Alarm
	ISO-8859-1	Success
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Alarm
	UTF8	Alarm
	UTF32	Alarm
	Other character sets	Alarm
GB2312	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Success
	GBK	Success
	GB18030	Success
	UTF16	Success
	UTF8	Success

Source Database	Target Database	Result
	UTF32	Success
	Other character sets	Alarm
GBK	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Success
	GB18030	Success
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
GB18030	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Success
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
UTF16	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF8	Success

Source Database	Target Database	Result
	UTF32	Success
	Other character sets	Alarm
UTF8	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
UTF32	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
ASCII	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Alarm
	UTF8	Alarm

Source Database	Target Database	Result
	UTF32	Alarm
	Other character sets	Alarm
UTF8MB3	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Success
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
UTF8MB4	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF8	Success
	UTF32	Success
	Other character sets	Alarm
UTF16LE	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF8	Success

Source Database	Target Database	Result
	UTF32	Success
	Other character sets	Alarm
Other character sets	SQL_ASCII	Alarm
	ISO-8859-1	Alarm
	LATIN1	Alarm
	GB2312	Alarm
	GBK	Alarm
	GB18030	Alarm
	UTF16	Alarm
	UTF8	Alarm
	UTF32	Alarm
	Other character sets	Alarm

Table 2-7 Character set compatibility between Oracle and MySQL

Source Database	Target Database	Result
SQL_ASCII	ASCII	Success
	LATIN1/ISO-8859-1	Alarm
	GB2312	Alarm
	GBK	Alarm
	UTF8MB3	Alarm
	GB18030	Alarm
	UTF16	Alarm
	UTF16LE	Alarm
	UTF8	Alarm
	UTF32	Alarm
	UTF8MB4	Alarm
	Other character sets	Alarm
US7ASCII	ASCII	Alarm
	LATIN1/ISO-8859-1	Alarm

Source Database	Target Database	Result
	GB2312	Alarm
	GBK	Alarm
	UTF8MB3	Alarm
	GB18030	Alarm
	UTF16	Alarm
	UTF16LE	Alarm
	UTF8	Alarm
	UTF32	Alarm
	UTF8MB4	Alarm
	Other character sets	Alarm
ISO-8859-1	ASCII	Alarm
	LATIN1/ISO-8859-1	Success
	GB2312	Alarm
	GBK	Alarm
	UTF8MB3	Alarm
	GB18030	Alarm
	UTF16	Alarm
	UTF16LE	Alarm
	UTF8	Alarm
	UTF32	Alarm
	UTF8MB4	Alarm
	Other character sets	Alarm
LATIN1	ASCII	Alarm
	LATIN1/ISO-8859-1	Success
	GB2312	Alarm
	GBK	Alarm
	UTF8MB3	Alarm
	GB18030	Alarm
	UTF16	Alarm
	UTF16LE	Alarm

Source Database	Target Database	Result
	UTF8	Alarm
	UTF32	Alarm
	UTF8MB4	Alarm
	Other character sets	Alarm
GB2312	ASCII	Alarm
	LATIN1/ISO-8859-1	Alarm
	GB2312	Success
	GBK	Success
	UTF8MB3	Success
	GB18030	Success
	UTF16	Success
	UTF16LE	Success
	UTF8	Success
	UTF32	Success
	UTF8MB4	Success
	Other character sets	Alarm
ZHS16GBK	ASCII	Alarm
	LATIN1/ISO-8859-1	Alarm
	GB2312	Alarm
	GBK	Success
	UTF8MB3	Success
	GB18030	Success
	UTF16	Success
	UTF16LE	Success
	UTF8	Success
	UTF32	Success
	UTF8MB4	Success
	Other character sets	Alarm
GBK	ASCII	Alarm
	LATIN1/ISO-8859-1	Alarm

Source Database	Target Database	Result
	GB2312	Alarm
	GBK	Success
	UTF8MB3	Success
	GB18030	Success
	UTF16	Success
	UTF16LE	Success
	UTF8	Success
	UTF32	Success
	UTF8MB4	Success
	Other character sets	Alarm
ZHS32GB18030	ASCII	Alarm
	LATIN1/ISO-8859-1	Alarm
	GB2312	Alarm
	GBK	Success
	UTF8MB3	Success
	GB18030	Success
	UTF16	Success
	UTF16LE	Success
	UTF8	Success
	UTF32	Success
	UTF8MB4	Success
	Other character sets	Alarm
GB18030	ASCII	Alarm
	LATIN1/ISO-8859-1	Alarm
	GB2312	Alarm
	GBK	Alarm
	UTF8MB3	Alarm
	GB18030	Success
	UTF16	Success
	UTF16LE	Success

Source Database	Target Database	Result
	UTF8	Success
	UTF32	Success
	UTF8MB4	Success
	Other character sets	Alarm
AL16UTF16	ASCII	Alarm
	LATIN1/ISO-8859-1	Alarm
	GB2312	Alarm
	GBK	Alarm
	UTF8MB3	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF16LE	Success
	UTF8	Success
	UTF32	Success
	UTF8MB4	Success
	Other character sets	Alarm
UTF16	ASCII	Alarm
	LATIN1/ISO-8859-1	Alarm
	GB2312	Alarm
	GBK	Alarm
	UTF8MB3	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF16LE	Success
	UTF8	Success
	UTF32	Success
	UTF8MB4	Success
	Other character sets	Alarm
AL32UTF8	ASCII	Alarm
	LATIN1/ISO-8859-1	Alarm

Source Database	Target Database	Result
	GB2312	Alarm
	GBK	Alarm
	UTF8MB3	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF16LE	Success
	UTF8	Success
	UTF32	Success
	UTF8MB4	Success
	Other character sets	Alarm
UTF8	ASCII	Alarm
	LATIN1/ISO-8859-1	Alarm
	GB2312	Alarm
	GBK	Alarm
	UTF8MB3	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF16LE	Success
	UTF8	Success
	UTF32	Success
	UTF8MB4	Success
	Other character sets	Alarm
UTF32	ASCII	Alarm
	LATIN1/ISO-8859-1	Alarm
	GB2312	Alarm
	GBK	Alarm
	UTF8MB3	Alarm
	GB18030	Alarm
	UTF16	Success
	UTF16LE	Success

Source Database	Target Database	Result
	UTF8	Success
	UTF32	Success
	UTF8MB4	Success
	Other character sets	Alarm
Other character sets	ASCII	Alarm
	LATIN1/ISO-8859-1	Alarm
	GB2312	Alarm
	GBK	Alarm
	UTF8MB3	Alarm
	GB18030	Alarm
	UTF16	Alarm
	UTF16LE	Alarm
	UTF8	Alarm
	UTF32	Alarm
	UTF8MB4	Alarm

2.3.3 Checking GUC Parameters

This check item is displayed when the target database type is GaussDB 3.1 or later and the source database type is Oracle, Microsoft SQL Server, or MySQL. Check whether GUC parameters are configured. For details, see **Table 2-8**.

NOTE

- The GUC parameter check is performed based on the target database version you selected during evaluation project creation, instead of the target database version that is actually connected.
- If the target database type is GaussDB Primary/Standby (M-compatible mode), GUC parameters cannot be checked.

Table 2-8 GUC parameter check items

Source DB Type	Target DB Version	Check Item
MySQL	GaussDB Primary/ Standby 3.1 Enterprise Edition	-

Source DB Type	Target DB Version	Check Item
	GaussDB Primary/ Standby 3.2 Enterprise Edition	b_format_behavior_compat_options
	GaussDB Distributed 3.2 Enterprise Edition	-
	GaussDB Primary/ Standby 3.3 Enterprise Edition	b_format_behavior_compat_options
	GaussDB Primary/ Standby 8.0 Enterprise Edition	b_format_version,b_format_dev_version
	GaussDB Distributed 8.0 Enterprise Edition	b_format_version,b_format_dev_version
	GaussDB Primary/ Standby 8.100 Enterprise Edition	b_format_version,b_format_dev_version
	GaussDB Distributed 8.100 Enterprise Edition	b_format_version,b_format_dev_version
Oracle	GaussDB Primary/ Standby 3.1 Enterprise Edition	behavior_compat_options,sql_beta_feature,a_format_version,a_format_dev_version
	GaussDB Distributed 3.1 Enterprise Edition	behavior_compat_options,sql_beta_feature,a_format_version,a_format_dev_version

Source DB Type	Target DB Version	Check Item
	GaussDB Primary/ Standby 3.2 Enterprise Edition	behavior_compat_options,plsql_compile_check_options,sql_beta_feature,a_format_version,a_format_dev_version
	GaussDB Distributed 3.2 Enterprise Edition	behavior_compat_options,sql_beta_feature,a_format_version,a_format_dev_version
	GaussDB Primary/ Standby 3.3 Enterprise Edition	behavior_compat_options,plsql_compile_check_options,sql_beta_feature,IntervalStyle,a_format_version,a_format_dev_version
	GaussDB Distributed 3.3 Enterprise Edition	behavior_compat_options,sql_beta_feature,IntervalStyle,a_format_version,a_format_dev_version
	GaussDB Primary/ Standby 8.0 Enterprise Edition	behavior_compat_options,plsql_compile_check_options,sql_beta_feature,IntervalStyle,a_format_version,a_format_dev_version
	GaussDB Distributed 8.0 Enterprise Edition	behavior_compat_options,sql_beta_feature,IntervalStyle,a_format_version,a_format_dev_version
	GaussDB Primary/ Standby 8.100 Enterprise Edition	behavior_compat_options,plsql_compile_check_options,sql_beta_feature,IntervalStyle,a_format_version,a_format_dev_version
	GaussDB Distributed 8.100 Enterprise Edition	behavior_compat_options,sql_beta_feature,IntervalStyle,a_format_version,a_format_dev_version
Microsoft SQL Server	GaussDB Primary/ Standby 3.1 Enterprise Edition	behavior_compat_options,sql_beta_feature,a_format_version,a_format_dev_version

Source DB Type	Target DB Version	Check Item
	GaussDB Distributed 3.1 Enterprise Edition	behavior_compat_options,sql_beta_feature,a_format_version,a_format_dev_version
	GaussDB Primary/ Standby 3.2 Enterprise Edition	behavior_compat_options,plsql_compile_check_options,sql_beta_feature,a_format_version,a_format_dev_version
	GaussDB Distributed 3.2 Enterprise Edition	behavior_compat_options,sql_beta_feature,a_format_version,a_format_dev_version
	GaussDB Primary/ Standby 3.3 Enterprise Edition	behavior_compat_options,plsql_compile_check_options,sql_beta_feature,a_format_version,a_format_dev_version
	GaussDB Distributed 3.3 Enterprise Edition	behavior_compat_options,sql_beta_feature,a_format_version,a_format_dev_version
	GaussDB Primary/ Standby 8.0 Enterprise Edition	behavior_compat_options,plsql_compile_check_options,sql_beta_feature,a_format_version,a_format_dev_version
	GaussDB Distributed 8.0 Enterprise Edition	behavior_compat_options,sql_beta_feature,a_format_version,a_format_dev_version
	GaussDB Primary/ Standby 8.100 Enterprise Edition	behavior_compat_options,plsql_compile_check_options,sql_beta_feature,a_format_version,a_format_dev_version
	GaussDB Distributed 8.100 Enterprise Edition	behavior_compat_options,sql_beta_feature,a_format_version,a_format_dev_version

2.3.4 Checking User Permissions

Separation of Permissions

Separation of permissions will be checked when the target database type is GaussDB. After separation of permissions is enabled, the system administrator (or a user with the SYSADMIN permission) does not have the CREATEROLE (security administrator) or AUDITADMIN (audit administrator) permissions, so the system administrator can neither create roles and users nor view or maintain database audit logs. After separation of permissions is enabled, a sysadmin user cannot migrate USER, ROLE, and GRANT objects.

NOTE

- This check item is displayed only when the target database type is GaussDB, separation of permissions is enabled, and the migration is performed by a sysadmin user.
- After separation of permissions is enabled, if a user with SYSADMIN permissions still has the security administrator (CREATEROLE) permissions and audit administrator permissions, this means that permission model is switched repeatedly. If you need to switch model from non-separation of permissions to separation of permissions, review the permissions of existing users and tailor some permissions as needed.

Sysadmin Permission

This check item is only displayed when the target database type is GaussDB, separation of permissions is disabled, and the migration user is granted the sysadmin permission. The check result must be **Success**.

Permission of Creating Schemas

This check item is displayed when the target database type is GaussDB and the migration user is granted the sysadmin permission. It is used to check whether the migration user has the permission to create schemas in the target database. Database objects must be created in schemas.

GRANT CREATE ON DATABASE *<db_name>* **TO** *<user>***;**

NOTE

When separation of permissions is enabled, initial user is used to grant permissions. When separation of permissions is disabled, SYSADMIN user is used to grant permissions.

This check item is mandatory.

Permissions of Creating and Modifying Users and Roles

This check item is displayed when the target database type is GaussDB, the migration user is not granted the sysadmin permission, and the USER, ROLE, and GRANT objects are migrated. It is used to check whether the migration user has the permissions to create or modify users and roles in the target DB instance.

ALTER USER *<user>* **WITH CREATEROLE;**

NOTE

- When separation of permissions is enabled, initial user is used to grant permissions. When separation of permissions is disabled, SYSADMIN user is used to grant permissions.
- This check item is mandatory.

Permission of Creating Objects Using PUBLIC SCHEMA

This check item is displayed when the target database type is GaussDB and the migration user is a common or sysadmin user (separation of permissions enabled). This check item involves complex permission combination.

In GaussDB 2.7 or earlier, CREATE permissions on public schemas are required to create objects such as tables, views, indexes, sequences, packages, types, and triggers.

In GaussDB 3.1 or later, the user SYSADMIN (used when separation of permission is enabled) must have the CREATE permissions on public schemas to create objects such as tables, views, indexes, sequences, packages, types, and triggers.

In GaussDB 3.1 or later, a common user must have the CREATE permissions on public schemas and the ANY permissions of objects, to create objects such as tables, views, indexes, sequences, packages, types, and triggers.

In separation of permissions, only initial users have the permissions to create functions, stored procedures, and synonyms on public schemas.

In non-separation of permissions, initial and sysadmin users have the permissions to create functions, stored procedures, and synonyms on public schemas.

CREATE permissions: **GRANT CREATE ON SCHEMA public TO** *<user>***;**

ANY permissions:

GRANT CREATE ANY TABLE TO *<user>;*//Users can create tables or views in public and user schemas.

GRANT CREATE ANY SEQUENCE TO *<user>***;**//Users can create sequences in public and user schemas.

GRANT CREATE ANY INDEX TO *<user>***;**//Users can create indexes in public and user schemas.

GRANT CREATE ANY PACKAGE TO *<user>***;**//Users can create packages in public and user schemas.

GRANT CREATE ANY TYPE TO *<user>***;**//Users can create types in public and user schemas.

GRANT CREATE ANY TRIGGER TO *<user>***;**//Users can create triggers in public and user schemas.

NOTE

- A DB instance contains multiple databases. Each database has its own public schema. Permission assignment must be performed in the corresponding database.
- When separation of permissions is enabled, initial user is used to grant permissions. When separation of permissions is disabled, SYSADMIN user is used to grant permissions.
- This check item is not mandatory. Based on the GaussDB permission design, the check result is always **Warning**.

Permissions of Existing Schemas

This check item is displayed when the target database type is GaussDB and the migration user is a common or sysadmin user (separation of permissions enabled).

It is used to check whether the migration user has the permissions to grant the owner of the existing schemas to the migration user.

GRANT *<schema_owner>* **TO** *<user>*

NOTE

When separation of permissions is enabled, initial user is used to grant permissions. When separation of permissions is disabled, SYSADMIN user or schema owner is used to grant permissions.

If the migration user does not have sufficient permissions, the schema owner cannot be queried.

Granting Tablespace Permissions

This check item is only displayed when the target database type is GaussDB, the migration user is not granted the sysadmin permission, and GRANT objects are migrated.

Statements:

GRANT <privilege> ON TABLESPACE <tablespace_name> TO <user>;

GRANT <privilege> ON TABLESPACE <tablespace_name> TO <user> WITH GRANT OPTION;

If **WITH GRANT OPTION** is specified, a grantee can grant this permission to others.

Permissions include CREATE, ALTER, DROP, COMMENT, CREATE WITH GRANT OPTION, ALTER WITH GRANT OPTION, DROP WITH GRANT OPTION and COMMENT WITH GRANT OPTION.

NOTE

When separation of permissions is enabled, initial user is used to grant permissions. When separation of permissions is disabled, SYSADMIN user is used to grant permissions.

Granting Database Permissions

This check item is only displayed when the target database type is GaussDB, the migration user is not granted the sysadmin permission, and GRANT objects are migrated.

Statements:

GRANT <privilege> ON DATABASE <db_name> TO <user>;

GRANT <privilege> ON DATABASE <db_name> TO <user> WITH GRANT OPTION;

If **WITH GRANT OPTION** is specified, a grantee can grant this permission to others.

Permissions include CREATE, CONNECT, TEMPORARY, ALTER, DROP, COMMENT, CREATE WITH GRANT OPTION, CONNECT WITH GRANT OPTION, ALTER WITH GRANT OPTION, TEMPORARY WITH GRANT OPTION, DROP WITH GRANT OPTION and COMMENT WITH GRANT OPTION.

NOTE

When separation of permissions is enabled, initial user is used to grant permissions. When separation of permissions is disabled, SYSADMIN user is used to grant permissions.

Granting ANY Permissions

This check item is displayed when the target database type is GaussDB, the migration user is not granted the sysadmin permission, and GRANT objects are migrated. It checks whether the migration user can grant ANY permissions on the target database to other users.

GRANT *<privilege>* **TO** *<user>* **WITH ADMIN OPTION;**

If **WITH ADMIN OPTION** is specified, the granted user can grant the permission to other roles or users.

Permissions include CREATE ANY TABLE, ALTER ANY TABLE, DROP ANY TABLE, SELECT ANY TABLE, UPDATE ANY TABLE, INSERT ANY TABLE, DELETE ANY TABLE, CREATE ANY SEQUENCE, ALTER ANY SEQUENCE, DROP ANY SEQUENCE, SELECT ANY SEQUENCE, CREATE ANY INDEX, ALTER ANY INDEX, DROP ANY INDEX, CREATE ANY FUNCTION, EXECUTE ANY FUNCTION, CREATE ANY PACKAGE, EXECUTE ANY PACKAGE, CREATE ANY TYPE, ALTER ANY TYPE, DROP ANY TYPE, CREATE ANY SYNONYM, DROP ANY SYNONYM, CREATE ANY TRIGGER, ALTER ANY TRIGGER and DROP ANY TRIGGER.

NOTE

A DB instance contains multiple databases. The ANY permissions are bound to databases. You need to grant the ANY permissions in the corresponding database.

When separation of permissions is enabled, initial user is used to grant permissions. When separation of permissions is disabled, SYSADMIN user is used to grant permissions.

Granting Permissions on the pg_catalog Schema

This check item is displayed when the target database type is GaussDB, the migration user is not granted the sysadmin permission, and GRANT objects are migrated. It checks whether the migration user can grant the query permission on all tables in the pg_catalog schema in the target database to other users.

GRANT *<privilege>* **TO** *<user>* **WITH ADMIN OPTION;**

NOTE

A DB instance contains multiple databases. Each database has its own pg_catalog schema. Permission assignment must be performed in the corresponding database.

When separation of permissions is enabled, initial user is used to grant permissions. When separation of permissions is disabled, SYSADMIN user is used to grant permissions.

Granting the Database Link Permission

This check item is displayed when the target database type is GaussDB, the migration user is not granted the sysadmin permission, and GRANT objects are migrated. It checks whether the migration user can grant the permission on creating database connections in the target database to other users.

GRANT CREATE PUBLIC DATABASE LINK TO *<user>* WITH GRANT OPTION;

NOTE

A DB instance contains multiple databases. You need to grant permissions in the corresponding database.

When separation of permissions is enabled, initial user is used to grant permissions. When separation of permissions is disabled, SYSADMIN user is used to grant permissions.

Granting Permissions to Check Role and User Authorization

This check item is displayed when the target database type is GaussDB, the migration user is not granted the sysadmin permission, and GRANT objects are migrated. It is used to check whether the migration user has the permissions to grant or modify other users' permissions.

NOTE

If a sysadmin user is created after separation of permissions is disabled, you can execute migration tasks as the user.

2.3.5 Setting Check Items

Setting a Character Set and Compatibility

Step 1 Connect to a source database and run the following SQL statements to query the character set type of schemas in the source database:

Oracle

```
SELECT value
FROM nls_database_parameters
WHERE parameter = 'NLS_CHARACTERSET';
```

MySQL

```
#databaseName indicates the name of a schema to be queried.
SELECT default_character_set_name FROM information_schema.schemata WHERE SCHEMA_NAME = databaseName;
```

Step 2 Create a compatible target database based on the character set compatibility rules in **Table 2-5** and **Table 2-7**.

CREATE DATABASE *DATABASENAME* ENCODING 'UTF8' DBCOMPATIBILITY = 'B' ;

NOTE

- You can replace **UTF8** with other character sets based on character compatibility rules.
- The **DBCOMPATIBILITY** parameter is used to set compatibility. For details, see **How Do I Create GaussDB Databases Compatible with Source Databases?**

----End

2.4 Converting Syntax

Converting DATE and TIME Functions

When you convert the syntax of target database GaussDB 3.2 or earlier, the DATE and TIME functions listed in **Table 2-9** return the transaction start time. The time you obtained in the source database is the current system time. In long transactions, the time obtained from the source database is different from that obtained from the target GaussDB database. If the converted statement contains these functions, you need to analyze whether there are long transactions and confirm the impact. If yes, manually modify the functions to ensure final accuracy.

Table 2-9 DATE and TIME functions

No.	Function
1	CURRENT_DATE
2	CURRENT_TIME
3	CURRENT_TIME(precision)
4	CURRENT_TIMESTAMP(precision)
5	LOCALTIME
6	LOCALTIMESTAMP
7	LOCALTIME(precision)
8	LOCALTIMESTAMP(precision)

Oracle -> GaussDB

GaussDB supports only common synonyms and does not support public synonyms. UGO will convert public synonyms in an Oracle database to common synonyms of the public schema in a GaussDB database. For details, see **Table 2-10**.

If different users in your database have public synonyms with the same name, there are synonym conflicts. You need to reconstruct the synonyms with the same name before migrating them. For details, see **Table 2-11**.

Table 2-10 Syntax conversion 1

Source SQL in Oracle	Converted SQL in GaussDB
CREATE PUBLIC SYNONYM u1.syn FOR u1.object1;	CREATE SYNONYM public.syn FOR u1.object;

Table 2-11 Syntax conversion 2

Source SQL in Oracle	Converted SQL in GaussDB	Whether Synonyms Conflict
CREATE PUBLIC SYNONYM u1.syn FOR u1.object1;	CREATE SYNONYM public.syn FOR u1.object1;	No
CREATE PUBLIC SYNONYM u2.syn FOR u1.object2;	CREATE SYNONYM public.syn FOR u2.object2;	Yes

Converting Syntax from PostgreSQL to GaussDB

Mode compatibility:

- Syntax can be migrated from PostgreSQL to GaussDB (O-compatible mode).
- In the GaussDB (O-compatible mode) database, an empty string is stored as null. If **True** is returned in the result after **select '' is null;** is executed, an empty string is also stored as null in the Oracle database.

Impact scope:

- If the source database type is PostgreSQL and the target database type is GaussDB (O-compatible mode), SQL statements containing empty character strings or null need to be reconstructed.
- The impact scope includes DML, DDL, and system functions, such as **select'' is null;** and **select regexp_matches('test1', 'test1', '');**.
- UGO can evaluate the IS NULL syntax and empty string syntax during migration from PostgreSQL to GaussDB.

Converting Syntax from Microsoft SQL Server to MySQL

The DATETIME data type of Microsoft SQL Server supports addition and subtraction operations. However, UGO cannot accurately identify this scenario, so the DATE_ADD function is recommended in the MySQL database for reconstruction.

2.5 Converting Data Types

2.5.1 From MySQL to GaussDB

Source Data Type	Converted Data Type (Versions Earlier Than GaussDB 8.0)	Converted Data Type (GaussDB 8.0 and Later Versions)	Conversion Configuration
CHAR	CHAR	CHAR	Character set length conversion If you select **This config will extend the character data type length**, the storage of characters converted by DRS and UGO is equivalent.
VARCHAR	VARCHAR	VARCHAR	Character set length conversion. If you select **This config will extend the character data type length**, the storage of characters converted by DRS and UGO is equivalent.
BINARY	BYTEA	BYTEA	-
VARBINARY	BYTEA	BYTEA	-
TINYBLOB	BYTEA	TINYBLOB	-
BLOB	BLOB	BLOB	BLOB If you select **BLOB data type will retain**, the storage of characters converted by DRS and UGO is equivalent.
MEDIUMBLOB	BYTEA	MEDIUMBLOB	-
LONGBLOB	BYTEA	LONGBLOB	LONGBLOB If you select **LONGBLOB will convert to BYTEA**, the storage of characters converted by DRS and UGO is equivalent.
TINYTEXT	TEXT	TINYTEXT	-
TEXT	TEXT	TEXT	TEXT datatype conversion If you select **Not conversion, retain the text data type, and comment out the length**, the storage of characters converted by DRS and UGO is equivalent.
MEDIUMTEXT	TEXT	MEDIUMTEXT	-

Source Data Type	Converted Data Type (Versions Earlier Than GaussDB 8.0)	Converted Data Type (GaussDB 8.0 and Later Versions)	Conversion Configuration
LONGTEXT	CLOB	LONGTEXT	LONGTEXT If you select **This config will convert LONGTEXT to CLOB**, the storage of characters converted by DRS and UGO is equivalent.
ENUM	VARCHAR	VARCHAR	-
SET	SET	SET	-
TINYINT	SMALLINT	TINYINT	TINYINT If you select **This data type will be converted to TINYINT.**, the storage of characters converted by DRS and UGO is equivalent. Check Constraint If you select **Do not add the check constraint**, the storage of characters converted by DRS and UGO is equivalent.
SMALLINT	SMALLINT	SMALLINT	UNSIGNED INTEGER If you select **The type remains unchanged**, the storage of characters converted by DRS and UGO is equivalent. Check Constraint If you select **Do not add the check constraint**, the storage of characters converted by DRS and UGO is equivalent.
MEDIUMINT	INTEGER	MEDIUMINT	SIGNED INTEGER If you select **The type remains unchanged**, the storage of characters converted by DRS and UGO is equivalent. UNSIGNED INTEGER If you select **The type remains unchanged**, the storage of characters converted by DRS and UGO is equivalent. Check Constraint If you select **Do not add the check constraint**, the storage of characters converted by DRS and UGO is equivalent.
INT	INT	INT	-
BIGINT	BIGINT	BIGINT	-

Source Data Type	Converted Data Type (Versions Earlier Than GaussDB 8.0)	Converted Data Type (GaussDB 8.0 and Later Versions)	Conversion Configuration
FLOAT	FLOAT4	FLOAT	FLOAT If you select **Convert type to DOUBLE PRECISION, comment out precision and scale, comment out sign and padding keywords (UNSIGNED, SIGNED, ZEROFILL)**, the storage of characters converted by DRS and UGO is equivalent. Check Constraint If you select **Do not add the check constraint**, the storage of characters converted by DRS and UGO is equivalent.
DOUBLE	DOUBLE PRECISION	DOUBLE	DOUBLE If you select **Convert type to DOUBLE PRECISION, comment out precision and scale, comment out sign and padding keywords (UNSIGNED, SIGNED, ZEROFILL)**, the storage of characters converted by DRS and UGO is equivalent. Check Constraint If you select **Do not add the check constraint**, the storage of characters converted by DRS and UGO is equivalent.
DATE	DATE	DATE	-
DATETIME	TIMESTAMP	DATETIME	-
TIMESTAMP	TIMESTAMP WITH TIME ZONE	TIMESTAMP	-
TIME	TIME	TIME	-
BIT	BIT	BIT	-
JSON	JSONB	JSONB	JSON data type conversion If you select **No conversion. Retain the JSON data type.**, the storage of characters converted by DRS and UGO is equivalent.
YEAR	SMALLINT	SMALLINT	Check Constraint If you select **Do not add the check constraint**, the storage of characters converted by DRS and UGO is equivalent.
DECIMAL	DECIMAL	DECIMAL	-

Source Data Type	Converted Data Type (Versions Earlier Than GaussDB 8.0)	Converted Data Type (GaussDB 8.0 and Later Versions)	Conversion Configuration
NUMERIC	NUMERIC	NUMERIC	-

2.5.2 From Oracle to GaussDB

Source Data Type	Converted Data Type	Conversion Configuration
CHAR	CHAR	Mismatch character set If you select **This config will adjust the size of CHAR(n), VARCHAR(n), VARCHAR2(n), and NCHAR(n) such that there will not be overflow**, the storage of characters converted by DRS and UGO is equivalent.
VARCHAR	VARCHAR	Mismatch character set If you select **This config will adjust the size of CHAR(n), VARCHAR(n), VARCHAR2(n), and NCHAR(n) such that there will not be overflow**, the storage of characters converted by DRS and UGO is equivalent.
VARCHAR2	VARCHAR2	Mismatch character set If you select **This config will adjust the size of CHAR(n), VARCHAR(n), VARCHAR2(n), and NCHAR(n) such that there will not be overflow**, the storage of characters converted by DRS and UGO is equivalent.
NCHAR	NCHAR	Mismatch character set If you select **This config will adjust the size of CHAR(n), VARCHAR(n), VARCHAR2(n), and NCHAR(n) such that there will not be overflow**, the storage of characters converted by DRS and UGO is equivalent.
NVARCHAR2	NVARCHAR2	-
NUMBER	NUMBER	-
NUMBER (6, 3)	NUMBER (6, 3)	-
NUMBER (6, 0)	NUMBER (6, 0)	-
NUMBER (3)	NUMBER (3)	-
NUMBER (6, -2)	NUMBER (6, -2)	-
BINARY_FLOAT	REAL	-
BINARY_DOUBLE	BINARY_DOUBLE	-

Source Data Type	Converted Data Type	Conversion Configuration
FLOAT	DOUBLE PRECISION	Float datatype If you select **The FLOAT/FLOAT(n) will be converted to DOUBLE PRECISION**, the storage of characters converted by DRS and UGO is equivalent.
INT	NUMBER(38)	Integer datatype If you select **This config will convert INT, INTEGER, and SMALLINT to NUMBER(38),** , the storage of characters converted by DRS and UGO is equivalent.
INTEGER	NUMBER(38)	Integer datatype If you select **This config will convert INT, INTEGER, and SMALLINT to NUMBER(38),** , the storage of characters converted by DRS and UGO is equivalent.
DATE	DATE	-
TIMESTAMP	TIMESTAMP(6)	-
TIMESTAMP_TZ	TIMESTAMP(6) WITH TIME ZONE	-
TIMESTAMP_LTZ	TIMESTAMP(6) WITH TIME ZONE	-
INTERVAL_YM	INTERVAL YEAR TO MONTH	-
INTERVAL_DS	INTERVAL DAY TO SECOND	-
BLOB	BLOB	-
CLOB	CLOB	-
NCLOB	TEXT	NCLOB datatype If you select **The NCLOB will be converted into TEXT datatype**, the storage of characters converted by DRS and UGO is equivalent.
LONG	TEXT	LONG data type If you select **The LONG type will be converted into TEXT data type**, the storage of characters converted by DRS and UGO is equivalent.
LONG_RAW	BYTEA	LONG RAW data type If you select **The LONG RAW will be converted into BYTEA data type**, the storage of characters converted by DRS and UGO is equivalent.
RAW	RAW	-

Source Data Type	Converted Data Type	Conversion Configuration
RowID	CHAR(18)	-
NUMBER(*, 0)	NUMBER(38, 0)	-

2.5.3 From PostgreSQL to GaussDB

Source Data Type	Converted Data Type	Conversion Configuration
SMALLINT	SMALLINT	-
INTEGER	INTEGER	-
BIGINT	BIGINT	-
INTEGER	INTEGER	-
REAL	REAL	-
DOUBLE PRECISION	DOUBLE PRECISION	-
NUMERIC	NUMERIC	-
CHARACTER VARYING	CHARACTER VARYING	Character datatype conversion If you select **Extended character data type char/varchar length**, the storage of characters converted by DRS and UGO is equivalent.
CHARACTER	CHARACTER	Character datatype conversion If you select **Extended character data type char/varchar length**, the storage of characters converted by DRS and UGO is equivalent.
BIT	BIT	-
BIT VARYING	BIT VARYING	-
BOOLEAN	BOOLEAN	-
BYTEA	BYTEA	-
TEXT	TEXT	-
TIME WITHOUT TIME ZONE	TIME WITHOUT TIME ZONE	-
TIME WITH TIME ZONE	TIME WITH TIME ZONE	-
TIMESTAMP WITHOUT TIME ZONE	TIMESTAMP WITHOUT TIME ZONE	-

Source Data Type	Converted Data Type	Conversion Configuration
TIMESTAMP WITH TIME ZONE	TIMESTAMP WITH TIME ZONE	-
INTERVAL	INTERVAL	-
CIDR	CIDR	-
PATH	PATH	-
BOX	BOX	-
LSEG	LSEG	-
MACADDR	MACADDR	-
POINT	POINT	-
POLYGON	POLYGON	-
INET	INET	-
TSQUERY	TSQUERY	-
TSVECTOR	TSVECTOR	-
UUID	UUID	-
JSON	JSON	-
JSONB	JSONB	-

2.5.4 From Microsoft SQL Server to GaussDB

Source Data Type	Converted Data Type	Conversion Configuration
TINYINT	TINYINT	-
SMALLINT	SMALLINT	-
INT	INT	-
BIGINT	BIGINT	-
DECIMAL	DECIMAL	-
NUMERIC	NUMERIC	-
FLOAT	FLOAT	-
REAL	REAL	-
SMALLMONEY	numeric(10, 4)	-
MONEY	numeric(19, 4)	-

Source Data Type	Converted Data Type	Conversion Configuration
BIT	BOOLEAN	Bit datatype If you select **Convert the bit type to boolean data type**, the storage of characters converted by DRS and UGO is equivalent.
DATE	DATE	-
SMALLDATETIME	SMALLDATETIME	-
DATETIME	TIMESTAMP(3)	-
DATETIME2(n)	TIMESTAMP(n)	-
DATETIMEOFFSET(n)	TIMESTAMPTZ(n)	datetimeoffset datatype If you select **Convert the datetimeoffset type to timestamptz**, the storage of characters converted by DRS and UGO is equivalent.
TIME(p)	TIME(p)	-
TIMESTAMP	BYTEA	-
XML	XML	XML datatype If you select **Convert XML data type to TEXT data type**, the storage of characters converted by DRS and UGO is equivalent.
CHAR(n)	CHAR(n)	Mismatch character set If you select **This config will adjust the size of CHAR(n), VARCHAR(n), and NCHAR(n) such that there will not be overflow**, the storage of characters converted by DRS and UGO is equivalent.
VARCHAR(n)	VARCHAR(n)	Mismatch character set If you select **This config will adjust the size of CHAR(n), VARCHAR(n), and NCHAR(n) such that there will not be overflow**, the storage of characters converted by DRS and UGO is equivalent.
varchar(max)	TEXT	VARCHAR(MAX) datatype If you select **Convert VARCHAR(MAX) data type to TEXT data type**, the storage of characters converted by DRS and UGO is equivalent.
NCHAR(n)	NCHAR(n)	Mismatch character set If you select **This config will adjust the size of CHAR(n), VARCHAR(n), and NCHAR(n) such that there will not be overflow**, the storage of characters converted by DRS and UGO is equivalent.
NVARCHAR(n)	NVARCHAR2(n)	-

Source Data Type	Converted Data Type	Conversion Configuration
NVARCHAR(max)	TEXT	NVARCHAR(MAX) datatype If you select **Convert NVARCHAR(MAX) data type to TEXT data type**, the storage of characters converted by DRS and UGO is equivalent.
BINARY(n)	BYTEA	-
VARBINARY(n)	BYTEA	-
VARBINARY(MAX)	BYTEA	VARBINARY(MAX) datatype If you select **Convert VARBINARY(MAX) data type to BYTEA data type**, the storage of characters converted by DRS and UGO is equivalent.
IMAGE	BYTEA	IMAGE datatype If you select **Convert IMAGE data type to BYTEA data type**, the storage of characters converted by DRS and UGO is equivalent.
HIERARCHYID	BYTEA	HIERARCHYID datatype If you select **Convert HIERARCHYID data type to BYTEA data type**, the storage of characters converted by DRS and UGO is equivalent.
NTEXT	TEXT	NTEXT datatype If you select **Convert NTEXT data type to TEXT data type**, the storage of characters converted by DRS and UGO is equivalent.
TEXT	TEXT	TEXT datatype If you select **Convert TEXT data type to TEXT data type**, the storage of characters converted by DRS and UGO is equivalent.
UNIQUEIDENTIFIER	UUID	-

3 SQL Statement Conversion

Scenarios

You can check whether a specific SQL statement can be converted in the source and target databases and view the SQL statement after conversion.

Prerequisites

The SQL text provided for UGO must be syntactically correct, can be successfully compiled on the source database, and contains up to 100,000 characters.

Frozen accounts cannot view or modify target database types or conversion configurations.

Procedure

Step 1 **Log in to the UGO console.**

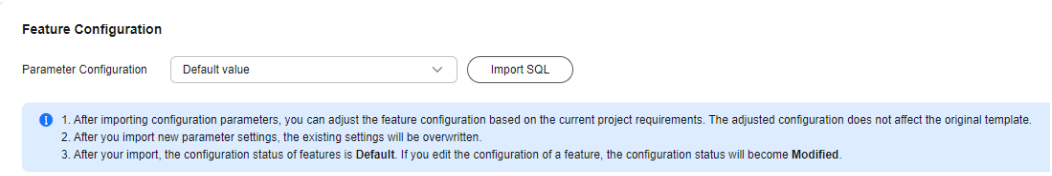
Step 2 In the navigation pane on the left, click **SQL Conversion**.

Step 3 Select the source and target database types and enter the SQL statements to be converted.

Step 4 (Optional) Click **Modify Conversion Configuration** to import the configuration parameters to be used. The conversion configuration in the default template is used by default.

- You can set **Parameter Configuration** to **Default value** or **Max compatibility**, or import a created conversion configuration template. For details about how to create a conversion configuration template, see **Creating a Configuration Template**.
- In the **Current Configuration** column, move the cursor to ⓘ of each feature to view the impact of the feature. You can click **Edit** to view details about configuration information and the current configuration conversion example.
- Locate a feature and click **Edit** in the **Operation** column to modify the current configuration of the feature, and then click **OK**.
- After you import configuration parameters, the configuration status of features is **Default value**. If you edit the configuration of a feature, the configuration status will become **Modified**.

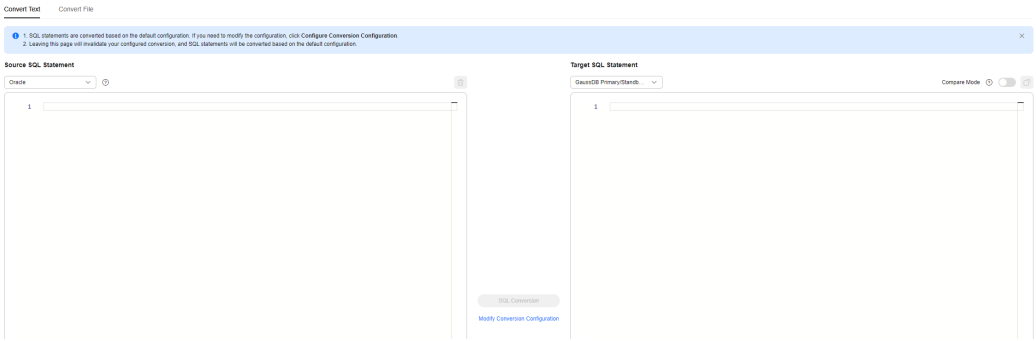
Figure 3-1 Feature configuration settings



Step 5 Click **SQL Conversion** and view the converted SQL statement on the right.

If any, the unsupported features are displayed in the **Unsupported Item** area in the lower right part.

Figure 3-2 SQL conversion



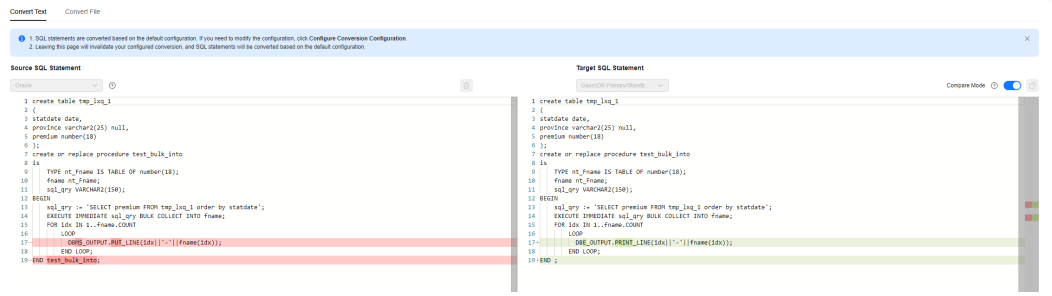
NOTE

- You can clear SQL statements on the source end and replicated SQL statements on the target end in a few clicks.
- For details about how to edit conversion configurations, see **Editing Conversion Configurations**.

Step 6 Enable **Compare Mode** (disabled by default).

After the source SQL statement is converted, enable **Compare Mode** to view the source SQL statement and converted SQL statement in comparison mode. In comparison mode, the source SQL statements cannot be modified.

Figure 3-3 Text comparison mode



----End

4 Conversion Configuration

4.1 Creating a Configuration Template

To use a conversion configuration solution to migrate multiple databases, you can create and set a conversion configuration template before creating a migration project. The template can be easily reused in subsequent migration projects.

Selecting a Template

Generally, you need to set **Template Type** to **Select** to create a conversion template.

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Conversion Configuration**.

Step 3 Click **Create Configuration Template**. **Template Type** is set to **Select** by default on the displayed page.

NOTE

- Each user can create up to 10 templates. If more than 10 templates have been created in an earlier version, all templates can be properly displayed after the upgrade.
- Only user **admin** can edit and delete templates created by other users.

Figure 4-1 Selecting a template

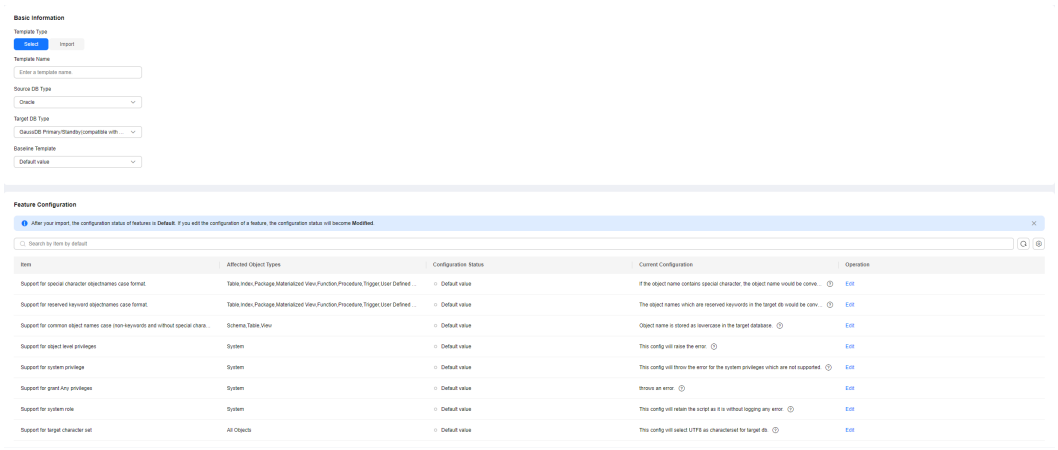


Table 4-1 Parameter description

Parameter	Description
Template Name	Enter a template name. The name is a string of 5 to 50 characters, consisting of only letters, digits, underscores (_), and hyphens (-). It must start with a letter (case insensitive) and end with a digit or letter.
Source DB Type	Select the type of the source database to be configured.
Target DB Type	Select the type of the target database to be configured.
Baseline Template	Select **Default value**, **Max compatibility**, or an existing template used for the same migration flow.

Step 4 You can select a preset configuration template from the **Baseline Template** drop-down list to quickly import conversion configurations. After you import configuration parameters, the configuration status of features is **Default value**. If you edit the configuration of a feature, the configuration status will become **Modified**.

WARNING

After you import new parameter settings, the existing settings will be overwritten. Exercise caution when performing this operation.

Step 5 Set the configuration items based on the actual conversion scenario.

- In the **Current Configuration** column, move the cursor to ⓘ of each feature to view the impact of the feature. You can click **View Sample** to view details about the configuration information and the current configuration conversion example.

- Locate a feature, click **Edit** in the **Operation** column to modify the current configuration, and then click **OK**.

Step 6 Click **Create**. You can view the created template in the template list.

----End

Importing a Template File

To import a conversion template file preset and exported by another account, select **Import** for **Template Type** and create a conversion template. For details about how to export a conversion configuration template, see **Exporting a Conversion Configuration Template**.

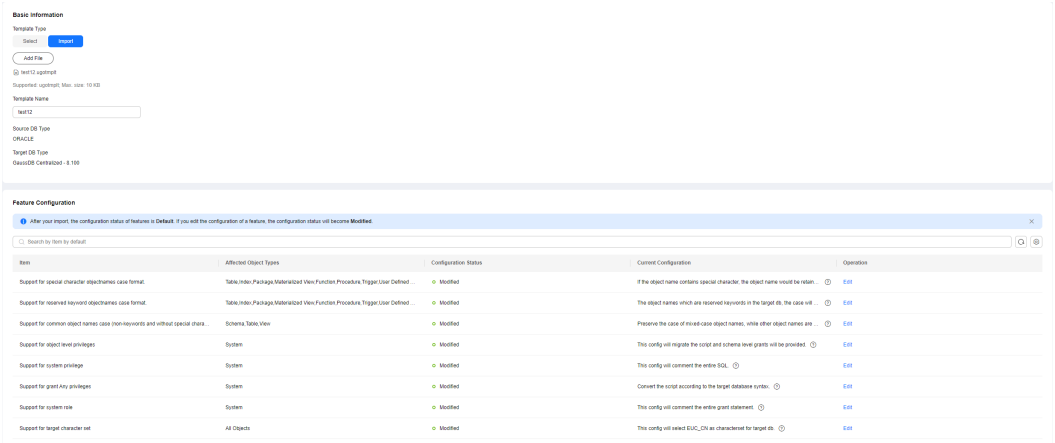
Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Conversion Configuration**.

Step 3 Click **Create Configuration Template**. On the displayed page, select **Import** for **Template Type**.

Step 4 On the displayed page, click **Add File** and upload the .ugotmplt template file exported from other instances. The system will identify and display the template name, source database type, target database type, and preset feature configurations.

Figure 4-2 Creating a configuration template by importing a template file



Step 5 (Optional) In the configuration item list, fine-tune the configuration based on the database to be migrated.

----End

4.2 Exporting a Conversion Configuration Template

Scenarios

If a configuration template has been created using the current Huawei Cloud account and needs to be reused by other accounts, you can export the template from the current UGO instance.

Prerequisites

You have created a template.

Procedure

Step 1 Log in to the UGO console.

Step 2 In the navigation pane, choose **Conversion Configuration**.

Step 3 Click **Export** in the **Operation** column to download the .ugotmplt template file.

----End

4.3 Viewing Conversion Configuration Templates

Scenarios

You can view custom conversion configuration templates, which are displayed in descending order of creation time.

Constraints

Only custom templates are displayed. System templates (including default values and maximum compatibility configurations) are not displayed.

Procedure

Step 1 Log in to the UGO console.

Step 2 In the navigation pane, choose **Conversion Configuration**.

Step 3 View the custom templates displayed in the list.

You can search for templates by template name, source database type, target database type, or creator. Fuzzy search is supported by template name and creator.

Figure 4-3 Viewing conversion configuration templates

----End

4.4 Modifying a Conversion Configuration Template

Procedure

Step 1 Log in to the UGO console.

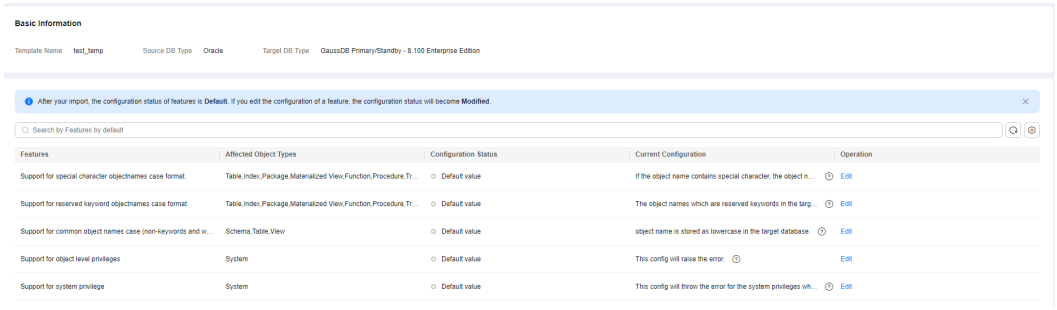
Step 2 In the navigation pane, choose **Conversion Configuration**.

Step 3 Locate a template that you want to modify and click **Edit** in the **Operation** column.

NOTE

- You can edit only the templates that you create.
- Only the value of a configuration item can be changed. The template name, source database type, and destination database type cannot be changed.
- After you import configuration parameters, the configuration status of features is **Default value**. If you edit the configuration of a feature, the configuration status will become **Modified**.
- Frozen accounts cannot be used to modify configuration templates.

Figure 4-4 Editing a template



Step 4 Locate a feature and click **Edit** in the **Operation** column to modify the current configuration of the feature.

Step 5 Click **OK**.

Step 6 After the modification, click **Back**.

----End

4.5 Deleting a Conversion Configuration Template

Scenarios

You can delete conversion configuration templates. Deleted templates cannot be restored. Exercise caution when performing this operation.

Prerequisites

You have created a template.

Templates frozen for security reasons cannot be deleted.

Procedure

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Conversion Configuration**.

Step 3 Locate a template that you want to delete and click **Delete** in the **Operation** column. You can also select multiple templates and click **Delete** above the list.

Step 4 Click **OK**.

----End

5 SQL Audit

SQL audit helps users detect SQL standardization, design rationality, and performance issues hidden in code at the development phase. More than 200 audit rules are preset for various SQL statements such as DML, DDL, and PL/SQL. You can adjust the risk level, threshold, and suggestion in a rule and create a custom audit template by combining multiple rules. GaussDB and MySQL databases can be audited. You can use a single statement, upload code files in batches (SQL statements are automatically extracted), or directly connect to the database to audit SQL statements. UGO also prevents inappropriate SQL statements from flowing into the production environment.

5.1 Statement Audit

5.1.1 Viewing Text Audit History

Scenarios

You can view details about the text to be audited.

Prerequisites

A text has been submitted for audit.

Procedure

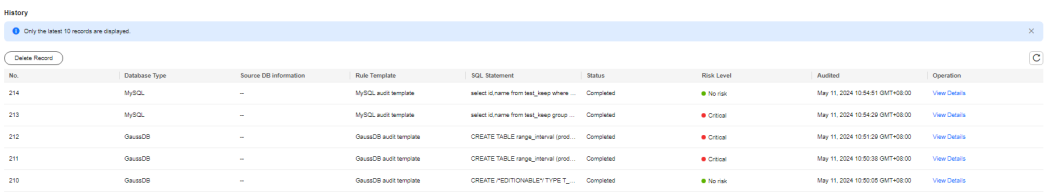
Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Statement Audit**. The **SQL Text** page is displayed by default.

Step 3 In the **History** area, view the historical records, including the database type, data source, SQL statement, and audit time.

Only the latest 10 records are displayed.

Figure 5-1 Viewing text audit history



Step 4 Click **Review Details**. The basic information, violation rules, and table structure of the SQL statement are displayed.

If the audit fails, the failure cause is displayed.

- **Violated Rules**: Based on the selected rule template, the system provides the violated rule name, risk level, and suggestions.

  Click the content in the **Problem Segment** column to highlight the problem segment in the text. To cancel the highlight, click the content again. The PL/SQL code may have multiple problem fragments. Click on the left to display all problem fragments.
- **Unmatched Rules**: The rules related to the source database are not matched because:
  - No data source is selected.
  - Data source connection fails.
  - The user permission is insufficient.
  - The schema is incorrectly selected.
  - The table is not found.

  The displayed information includes the rule name, risk level, and unmatched reason.
- **Table Structure**: Based on the selected rule template, the system provides the table structure information that the SQL statement to be audited depends on, including the object name, object type, and owner.
- **Execution Plans**: The execution plans of the SELECT, UPDATE, DELETE, and INSERT statements are displayed.
- Click the value next to **Rule Template** to go to the specific template information.

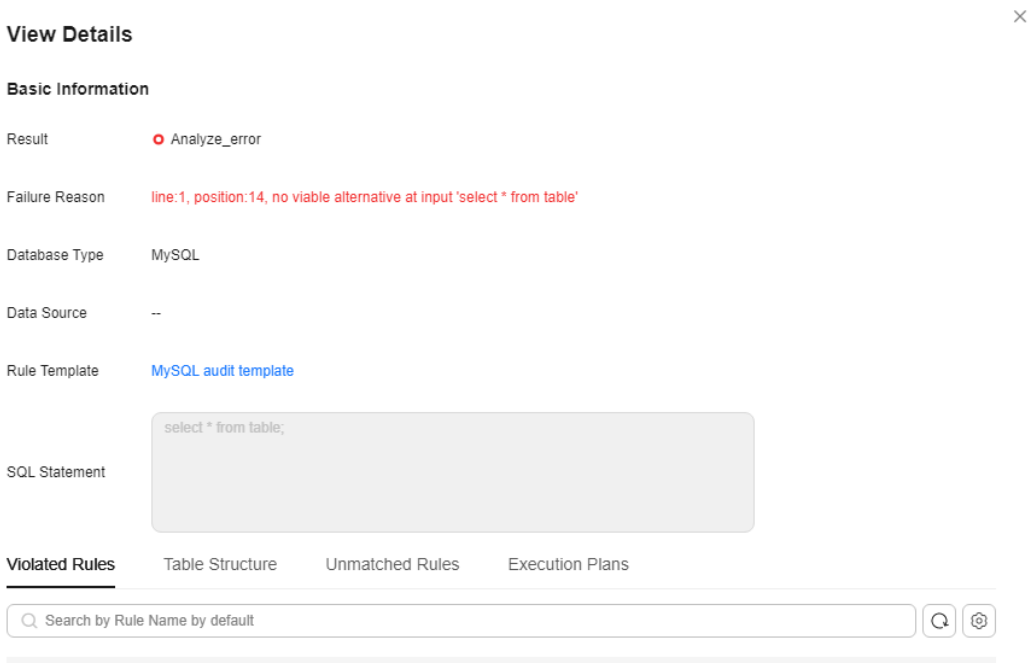
  NOTE

  To obtain the table structure, you must have the permissions to read system catalogs and system views.

  GaussDB system catalogs: tables, pg_class, pg_namespace, relnamespace, pgxc_class, pg_settings, pg_get_viewdef, columns, pg_tables, pg_index, table_constraints, pg_partition, pg_get_tabledef, check_constraints, key_column_usage, pg_get_keywords, gs_package, db_constraints, and adm_ind_columns.

  MySQL system catalogs: information_schema.tables, information_schema.columns, information_schema.statistics, and information_schema.table_constraints.

Figure 5-2 Viewing details



----End

5.1.2 Deleting All Audit Records

Scenarios

You can delete all audit records at a time. Deleted records cannot be restored. Exercise caution when performing this operation.

Prerequisites

A text has been submitted for audit.

Procedure

Step 1 Log in to the UGO console.

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Statement Audit**. The **SQL Text** page is displayed by default.

In the **History** area, view the historical records.

Step 3 Click **Delete Record** above the list.

Step 4 Click **OK** to clear all the records.

----End

5.1.3 Viewing File Details

Scenarios

You can view details about the uploaded SQL files.

Prerequisites

A file has been uploaded.

Procedure

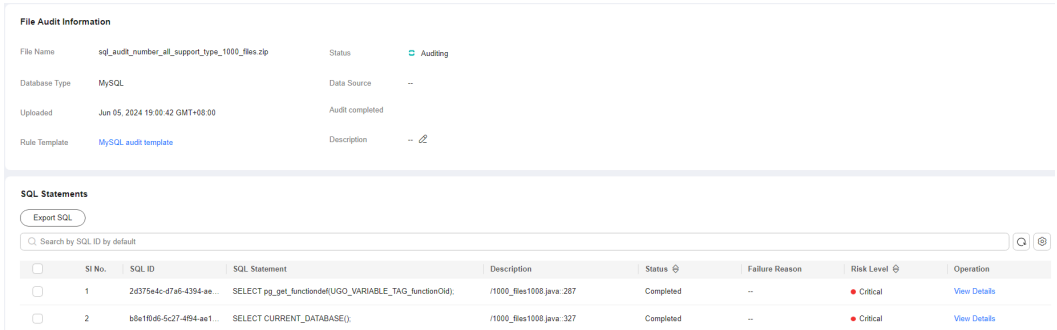
Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Statement Audit**. Click the **SQL from Files** tab.

Step 3 Locate a task and click **View Details** in the **Operation** column to view the SQL file audit information and collected SQL statements.

- **Export SQL**: Select one or more SQL statements to be exported and click **Export SQL** to export them to a **.tar.gz** file.
  - By default, all SQL statements are exported. You can also select SQL statements to export, or set search criteria to export desired ones.
  - By default, **The exported Excel file contains existing SQL statements** is selected. If the existing SQL statements are too large, do not select this option.
  - If the existing SQL statements in the exported file are too large, save them in a separate file whose name will be displayed in the **SQL Statement** column in the Excel file.
  - You can export a maximum of 5,000 objects to an Excel file.
  - If the table structure information which the SQL statement to be audited depends on contains more than 30,000 characters, it is exported as a SQL file.
- In the **File Audit Information** area, you can view the file name, audit status, and data source information.
- In the **SQL Statements** area, you can view SQL ID, SQL statement, description, risk level, and status.

Figure 5-3 Audit details

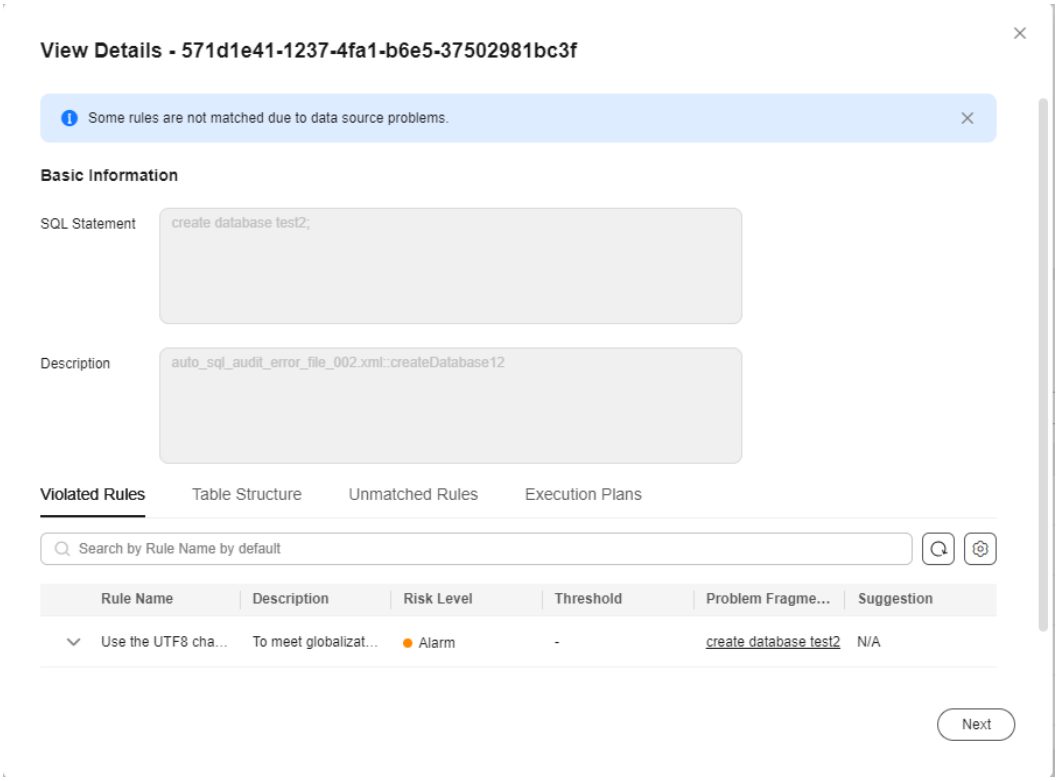


- Click the value next to **Rule Template** to view the specific template information.
- The status can be:
  - **New**: The collected SQL statement is waiting for audit.
  - **Completed**: The audit is complete.
  - **Analyze_error**: Failed to parse SQL statements.
  - **Error**: An exception occurred during the audit.
  - **Ignore**: The SQL statement cannot be audited or the size of a single SQL statement exceeds the threshold. The default threshold is 100 KB.
- Parsing failure cause:
  - **line:1, position:14, token:table** indicates the SQL statement contains **table**.
  - **line:1, position:3, token:<EOF>** indicates that the SQL statement is incomplete.

Step 4 Locate a SQL statement and click **View Details** in the **Operation** column to display the basic information, violated rules, and table structure of the SQL statement.

- If multiple SQL statements are collected, you can click **Previous** or **Next**.

Figure 5-4 Viewing SQL audit details



- **Basic Information**: The specific SQL statement and description are displayed.
- **Violated Rules**: Based on the selected rule template, the system provides the violated rule name, risk level, and suggestions.

  Click the content in the **Problem Segment** column to highlight the problem segment in the text. To cancel the highlight, click the content again. The

PL/SQL code may have multiple problem fragments. Click on the left to display all problem fragments.

- **Unmatched Rules**: The rules related to the source database are not matched because: no data source is selected, the data source connection fails, the user permission is insufficient, the schema is incorrectly selected, or the table is not found. The displayed information includes the rule name, risk level, and unmatched reason.
- **Table Structure**: Based on the selected rule template, the system provides the table structure information that the SQL statement to be audited depends on, including the object name, object type, and owner.
- **Execution Plans**: The execution plans of the SELECT, UPDATE, DELETE, and INSERT statements are displayed.

  NOTE

  To obtain the table structure, you must have the permissions to read system catalogs and system views.

  GaussDB system catalogs: tables, pg_class, pg_namespace, relnamespace, pgxc_class, pg_settings, pg_get_viewdef, columns, pg_tables, pg_index, table_constraints, pg_partition, pg_get_tabledef, check_constraints, key_column_usage, pg_get_keywords, gs_package, db_constraints, and adm_ind_columns.

  MySQL system catalogs: information_schema.tables, information_schema.columns, information_schema.statistics, and information_schema.table_constraints.

----End

5.1.4 Viewing Error Files

Scenarios

You can view the error files and error information during the audit.

Prerequisites

A file has been uploaded.

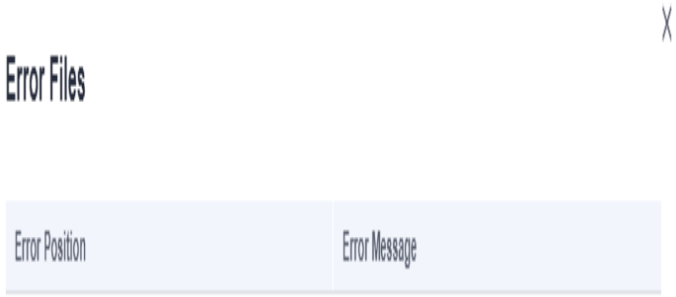
Procedure

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Statement Audit**. Click the **SQL from Files** tab.

Step 3 Locate the target task and choose **More** > **View Error Files**.

Figure 5-5 Viewing error files



----End

5.1.5 Deleting a File Audit Task

Scenarios

You can delete file audit tasks. Deleted audit tasks cannot be restored. Exercise caution when performing this operation.

Prerequisites

A file has been uploaded.

Procedure

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Statement Audit**. Click the **SQL from Files** tab.

Step 3 Locate the task you want to delete and choose **More** > **Delete** in the **Operation** column.

Step 4 In the displayed dialog box, click **OK**.

----End

5.2 Database Audit

5.2.1 Querying Audit Task Details

Scenarios

You can view details about a database audit task.

Prerequisites

A database audit task has been created

Procedure

Step 1 Log in to the UGO console.

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Database Audit**.

Step 3 Locate a task, and click **View Details** in the **Operation** column to view the audit details, object statistics, SQL risk statistics, and SQL statements.

Step 4 View information on the **Database Audit Details** area.

Figure 5-6 Viewing database audit details

- Click the value next to **Rule Template** to view the specific template information.
- To modify the task description, click next to **Description**.

Step 5 View information on the **Object Type Statistics** area.

Figure 5-7 Viewing object type statistics

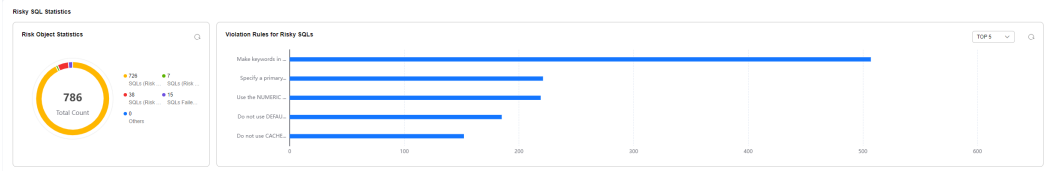
NOTE

The statistics are displayed only when the task status is **Audit completed**.

- **Object Statistics**: The number of object types to be audited in the database are displayed.
- **Object Types Involved in Risky SQLs that Violate Rules**: Objects that violate rules are displayed. You can move the cursor to the bar chart to view the number of objects.
  - You can click the drop-down list box in the upper right corner to view the first 5 or 10 object types that violate rules. By default, the first 5 objects are displayed.

Step 6 View information on the **Risky SQL Statistics** area.

Figure 5-8 Viewing risky SQL statistics



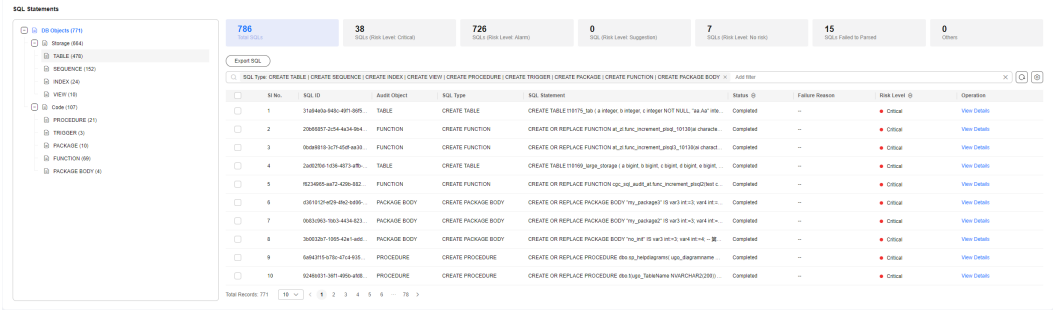
NOTE

Others refer to statements that cannot be parsed.

- **Risk Object Statistics**: Total number of risky objects in the audit database is collected.
- **Violation Rules for Risky SQLs**: Rules that are violated in DDL statements are displayed. You can move the cursor to the bar chart to view the number of rules.
  - You can click the drop-down list box in the upper right corner to view the first 5 or 10 rules that are violated. By default, the first 5 rules are displayed.

Step 7 View information on the **SQL Statements** area.

Figure 5-9 Viewing SQL statements



View the audited object types in the left navigation tree. After you click an object type, the **Object Statistics** and **Risky SQL Statistics** areas show the object type details.

- **Export SQL**: Select one or more SQL statements to be exported and click **Export SQL** to export them to a **.tar.gz** file.
  - By default, all SQL statements are exported. You can also select SQL statements to export, or set search criteria to export desired ones.
  - By default, **The exported Excel file contains existing SQL statements** is selected. If the existing SQL statements are too large, do not select this option.
  - If the existing SQL statements in the exported file are too large, save them in a separate file whose name will be displayed in the **SQL Statement** column in the Excel file.
  - You can export a maximum of 5,000 objects to an Excel file.
  - If the table structure information which the SQL statement to be audited depends on contains more than 30,000 characters, it is exported as a SQL file.
- The status can be:
  - **New**: The object SQL statement has just been collected and is waiting for audit.
  - **Completed**: The audit is complete.
  - **Analyze_error**: Failed to parse SQL statements.

  - **Error**: An exception occurred during the audit.
  - **Ignore**: The SQL statement cannot be audited or the size of a single SQL statement exceeds the threshold. The default threshold is 100 KB.

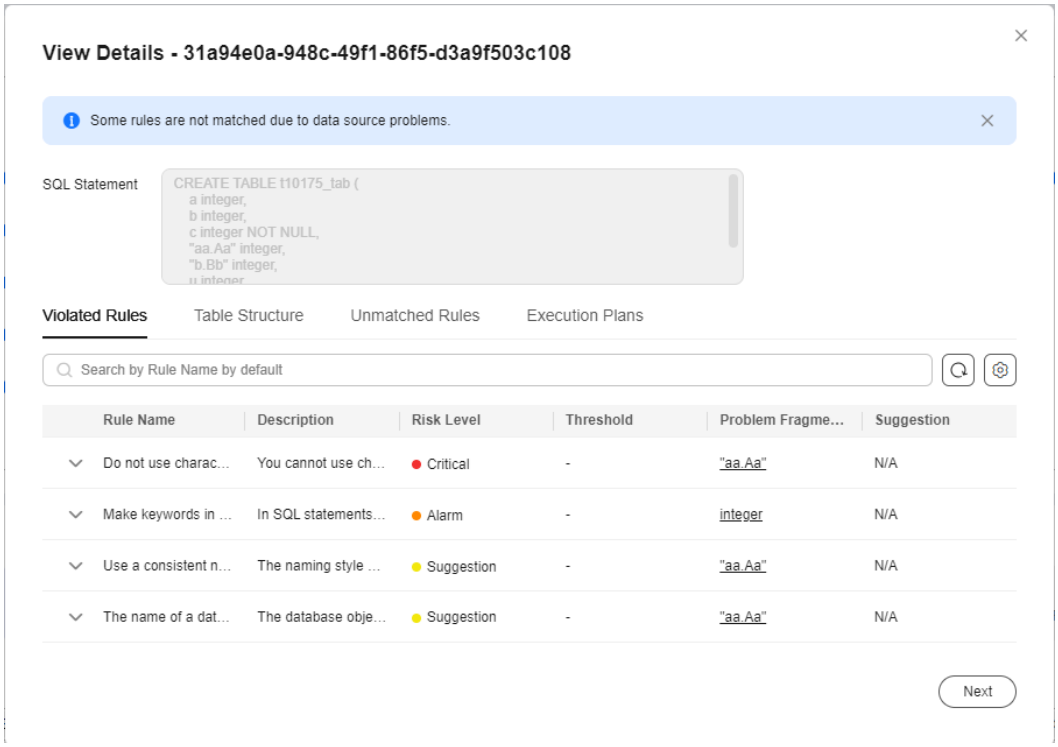
Failure cause:

- **line:1, position:14, token:table** indicates the SQL statement contains **table**.
- **line:1, position:3, token:<EOF>** indicates that the SQL statement is incomplete.

Step 8 Locate a SQL statement and click **View Details** in the **Operation** column. The basic information, violated rules, and table structure of the SQL statement are displayed.

- If multiple SQL statements are collected, you can click **Previous** or **Next**.

Figure 5-10 Viewing SQL audit details



- **SQL Statement**: The SQL statements for object creation are displayed.
- **Violated Rules**: Based on the selected rule template, the system provides the violated rule name, risk level, and suggestions.

  Click the content in the **Problem Segment** column to highlight the problem segment in the text. To cancel the highlight, click the content again. The PL/SQL code may have multiple problem fragments. Click on the left to display all problem fragments.
- **Table Structure**: Based on the selected rule template, the system provides the table structure information that the SQL statement to be audited depends on, including the object name, object type, and owner.
- **Execution Plans**: The execution plans of the SELECT, UPDATE, DELETE, and INSERT statements are displayed.

NOTE

To obtain the table structure, you must have the permissions to read system catalogs and system views.

- GaussDB system catalogs: tables, pg_class, pg_namespace, relnamespace, pgxc_class, pg_settings, pg_get_viewdef, columns, pg_tables, pg_index, table_constraints, pg_partition, pg_get_tabledef, check_constraints, key_column_usage, pg_get_keywords, db_constraints, and adm_ind_columns.
- Currently, table structures supported by index and view objects can be viewed.

----End

5.2.2 Deleting an Audit Task

Scenarios

You can delete an audit task. Deleted tasks cannot be restored. Exercise caution when performing this operation.

Prerequisites

An audit task has been created.

Procedure

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Database Audit**.

Step 3 Locate the task you want to delete and choose **More** > **Delete** in the **Operation** column.

Step 4 In the displayed dialog box, click **OK**.

----End

5.3 Rules

5.3.1 Viewing Rule Template Details

Scenarios

You can only view the details of each rule in system templates, but you can modify the threshold and severity of a single rule in a custom template as needed.

Prerequisites

There is a system template or custom template.

Procedure

Step 1 Log in to the UGO console.

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Rules**. The **Templates** page is displayed by default.

Step 3 Locate a template and click **View Details** in the **Operation** column.

- System templates:

  You can only view the template details, including the rule name, description, applicable database, and audited object types, but cannot modify them. You can click **Previous** or **Next** to view the information.

- Custom templates: You can add, modify, or delete rules in the templates.

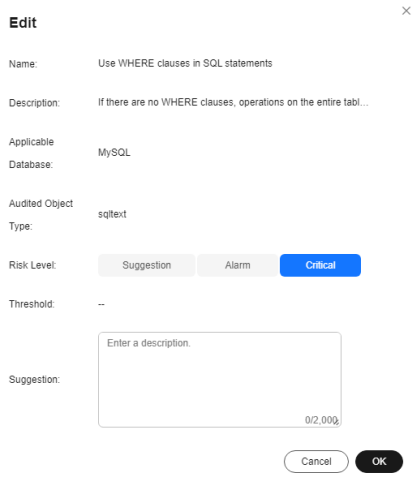
  To modify the template description, click next to **Description**. There are up to 100 characters.

  To modify a rule, click **Edit** in the **Operation** column to modify the rule risk level, threshold, and suggestion.

  Search for rules by rule name or rule ID.

Step 4 (Optional) Locate a rule of the custom template and click **Edit** in the **Operation** column.

Figure 5-11 Editing an existing rule in a custom template



Step 5 Modify risk level, threshold, and suggestion of the rule, and click **OK**.

NOTE

- In the custom template details, the projects in which the current template has been applied are displayed. After the modification is complete, new threshold is applied.
- **Threshold**:
  - The value can contain up to 100 characters. The value ranges from **0** to **2147483647**.
  - The value can contain nine decimal places. If the threshold is of the decimal type and does not contain the decimal part, the decimal point cannot be omitted, for example, **3.0**. If the threshold is an integer, the decimal point cannot be added, for example, **3**.
- **Suggestion**: There are up to 2,000 characters.

----End

Adding a Rule

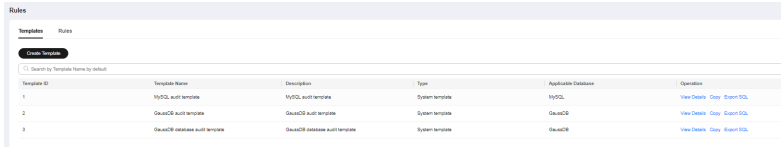
Step 1 Log in to the UGO console.

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Rules**. The **Templates** page is displayed by default.

Step 3 Locate a custom template and click **View Details** in the **Operation** column.

Step 4 Click **Add Rule**. The rules that are not contained in the template are displayed.

Figure 5-12 Adding a rule



Step 5 Select the required rules and click **OK**.

----End

Deleting a Rule

Step 1 Log in to the UGO console.

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Rules**. The **Templates** page is displayed by default.

Step 3 Locate a custom template and click **View Details** in the **Operation** column.

Step 4 Locate the rule to be deleted and click **Delete** in the **Operation** column.

Step 5 Click **OK**.

----End

5.3.2 Copying a Rule Template

Scenarios

You can copy an existing template to create a custom template.

Prerequisites

There is a rule template.

Procedure

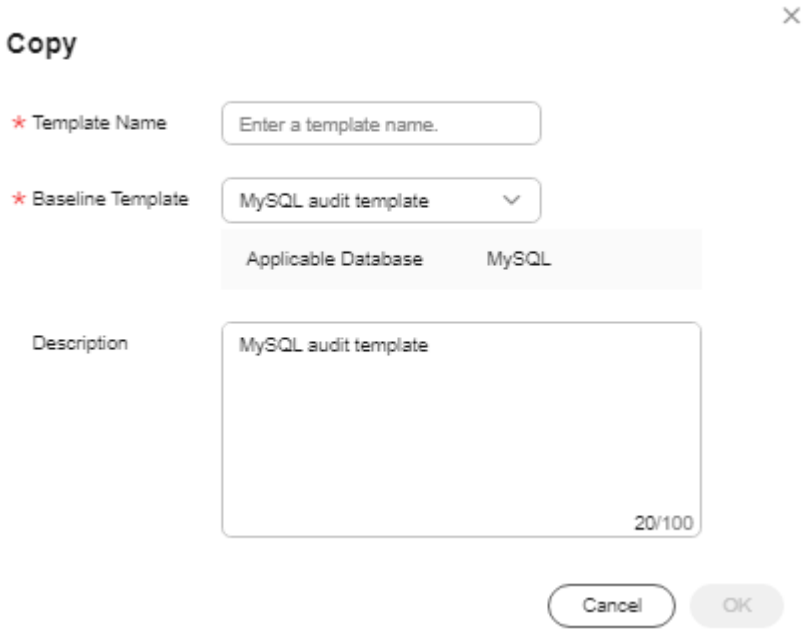
Step 1 Log in to the UGO console.

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Rules**. The **Templates** page is displayed by default.

Step 3 Select the template to be copied and click **Copy** in the **Operation** column.

The template name must be unique and cannot be empty. The baseline template is the template to be copied.

Figure 5-13 Copying a template



Step 4 Click **OK**. The copied template is displayed in the template list.

----End

5.3.3 Deleting a Rule Template

Scenarios

You can delete custom templates. Deleted templates cannot be restored. Exercise caution when performing this operation.

Prerequisites

There is a custom template.

Procedure

Step 1 Log in to the UGO console.

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Rules**. The **Templates** page is displayed by default.

Step 3 Locate a custom template to be deleted and click **Delete** in the **Operation** column.

Step 4 In the displayed dialog box, click **OK**.

----End

5.3.4 Viewing Rule Details

Scenarios

You can view existing system rules.

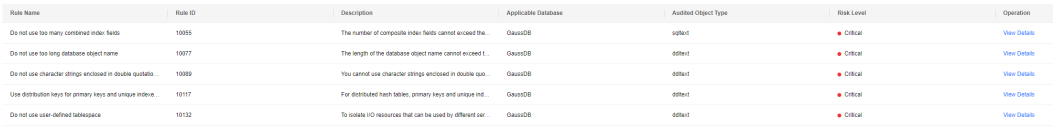
Procedure

Step 1 Log in to the UGO console.

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Rules**. Click the **Rules** tab.

Step 3 All existing system rules of GaussDB and MySQL are displayed in descending order of risk level. GaussDB rules are followed by MySQL rules.

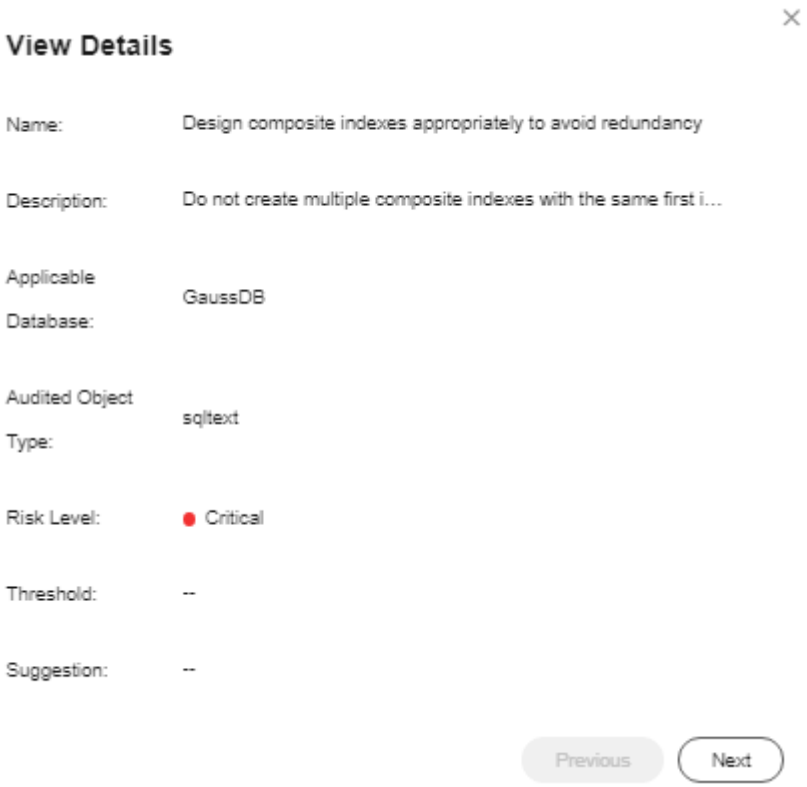
Figure 5-14 Viewing all rules



Step 4 Locate a rule and click **View Details** in **Operation** column.

You can click **Previous** or **Next** to view rules as needed.

Figure 5-15 Viewing details of a rule



----End

5.3.5 Exporting a Rule Template

Scenarios

You can export a rule template based on service requirements, modify the threshold and risk level in the exported Excel template, and upload the template again.

Procedure

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane on the left, choose **SQL Audit** > **Rules**. The **Templates** page is displayed by default.

Step 3 In the list, select the template to be exported and click **Export** in the **Operation** column to export the rule template to the local PC.

----End

6 Data Source Management

6.1 Testing Data Source Connection

Scenarios

You can check whether a data source is properly connected.

Prerequisites

At least one data source has been created.

Procedure

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Data Source Management**.

Step 3 In the data source management list, locate the data source to be tested and click **Test Connection** in the **Operation** column.

- If the test is successful, a message is displayed, indicating that the connection is successful.
- If the test fails, a message is displayed, indicating that the connection fails.

NOTE

If a data source is being tested, its **Test Connection** and **Delete** buttons are unavailable.

----End

6.2 Deleting a Data Source

Deleted data sources cannot be recovered. Exercise caution when performing this operation.

Prerequisites

A data source has been created and is not in the test connection state.

Procedure

Step 1 **Log in to the UGO console.**

Step 2 In the navigation pane, choose **Data Source Management**.

Step 3 In the data source management list, locate the data source to be deleted and click **Delete** in the **Operation** column.

Step 4 In the displayed dialog box, click **OK**.

----End

7 Permissions Management

7.1 Creating a User and Granting Permissions

This section describes how to use **IAM** to implement fine-grained permissions control for your UGO resources. With IAM, you can:

- Create IAM users for employees based on your enterprise's organizational structure. Each IAM user will have their own security credentials for accessing UGO resources.
- Grant only the permissions required for users to perform specific tasks.
- Entrust a Huawei Cloud account or cloud service to perform efficient O&M on your UGO resources.

If your Huawei Cloud account does not require individual IAM users, skip this section.

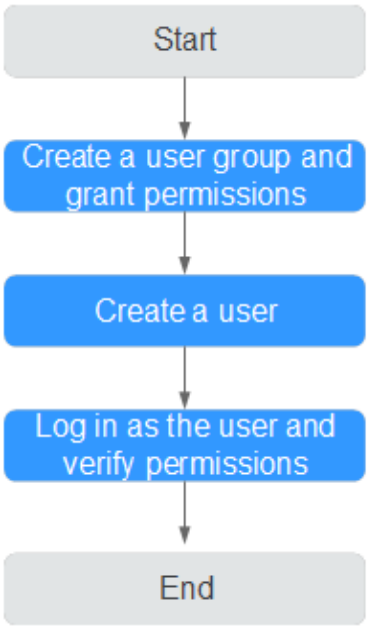
Figure 7-1 describes the process for granting permissions.

Prerequisites

Before assigning permissions to user groups, you should learn about the system-defined roles and policies listed in **Supported system roles**. For the system policies of other services, see **System Permissions**.

Process Flow

Figure 7-1 Process for granting UGO permissions



1. **Create a user group and assign permissions**.

   Create a user group on the IAM console, and assign the UGO ReadOnlyAccess policy to the group.

2. **Create an IAM user**.

   Create a user on the IAM console and add the user to the group created in **1**.

3. **Log in as an IAM user** and verify permissions.

   Log in to the management console using the newly created user, and verify that the user only has read permissions for UGO.

   In the service list, choose **Databases** > **Database and Application Migration UGO**. On the UGO page, choose **Schema Migration** > **Database Evaluation**, and click **Create Project** to create an evaluation project. If the evaluation project can be created (assume that the current permission contains only UGO Administrator), the evaluation project is UGO Administrator. The permission has taken effect.

7.2 Creating a Custom Policy

You are advised to use system-defined policies. If you need to create custom policies, see **Permission Management**.

You can create a custom policy in either of the following ways:

- Visual editor: Select cloud services, actions, resources, and request conditions. This does not require knowledge of policy syntax.
- JSON: Create a policy in JSON format or edit the JSON strings of an existing policy.

For details, see **Creating a Custom Policy**. This section contains examples of common UGO custom policies.

Example Custom Policies

- Example 1: Allowing a user to create an evaluation project

```
{
   "Version": "1.1",
   "Statement": [{
      "Action": ["ugo:evaluationJob:create"]
            "Effect": "Allow",
   }]
}
```

- Example 2: Denying a user to delete an evaluation project

  A policy with only "Deny" permissions must be used in conjunction with other policies to take effect. If the policies assigned to a user contain both "Allow" and "Deny", the "Deny" permissions take precedence over the "Allow" permissions.

  The following method can be used if you need to assign permissions of the **UGO FullAccess** policy to a user but you want to prevent the user from deleting UGO instances. Create a custom policy for denying evaluation project deletion, and attach both policies to the group to which the user belongs. Then, the user can perform all operations on evaluation projects except deleting evaluation projects. The following is an example of a deny policy:

```
{
   "Version": "1.1",
   "Statement": [{
      "Action": ["ugo:evaluationJob:delete"],
      "Effect": "Deny"
   }]
}
```


8 Interconnection with CTS

8.1 Key Operations Recorded by CTS

Cloud Trace Service (CTS) records operations on UGO resources for query, audit, and backtrack.

Table 8-1 UGO operations supported by CTS

Operation	Resource Type	Trace Name
Creating a database evaluation project	evaluate	createEvaluationJob
Creating an object migration project	migrate	createMigrationJob
Testing connection for the database evaluation project	evaluate	testConnection
Testing connection for an object migration project	migrate	testConnection
Deleting a database evaluation project	evaluate	deleteEvaluationJob
Deleting an object migration project	migrate	deleteMigrationJob
Starting object migration verification	migrate	startVerify
Stopping object migration verification	migrate	stopVerify
Starting object migration	migrate	startMigrate
Stopping object migration	migrate	stopMigrate

Operation	Resource Type	Trace Name
Stopping a database evaluation project	evaluate	stopEvaluationJob
Resuming a database evaluation project	evaluate	resumeEvaluationJob
Confirming the target database	evaluate	confirmEvaluationJob
Re-evaluating the database	evaluate	reanalyzeEvaluation
Performing a pre-check	evaluate	preCheck
Uploading a certificate	evaluate	uploadCertFile
Testing the network stability	evaluate	testNetwork

8.2 Viewing Traces

After CTS is enabled, CTS starts recording operations on cloud resources. You can view the operation records of the last seven days on the CTS console.

This section describes how to view operation records for the last 7 days on the CTS console.

Prerequisites

The CTS service has been enabled.

Procedure

Step 1 Log in to the console.

Step 2 Click  on the upper left to select the desired region and project.

Step 3 Click **Service List**. Under **Management & Governance**, choose **Cloud Trace Service**.

Step 4 Choose **Trace List** in the navigation pane on the left.

Step 5 Click **Filter** and specify filter criteria as needed. The following filters are available:

- **Trace Source**, **Resource Type**, and **Search By**

  Select a filter from the drop-down list.

  When you select **Trace name** for **Search By**, you also need to select a specific trace name.

  When you select **Resource ID** for **Search By**, you also need to select or enter a specific resource ID.

When you select **Resource name** for **Search By**, you also need to select or enter a specific resource name.

- **Operator**: Select a specific operator (a user rather than a tenant).
- **Trace Status**: Available options include **All trace statuses**, **Normal**, **Warning**, and **Incident**. You can only select one of them.
- Start time and end time: You can specify a time period for querying traces.

Step 6 Click ⌄ on the left of the target record to expand its details.

Step 7 Click **View Trace** in the **Operation** column. On the displayed dialog box, the trace structure details are displayed.

----End

