

TDSQL-C for MySQL

Best Practices

Product Documentation



Copyright Notice

©2013-2024 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

Trademark Notice



All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

Contents

Best Practices

Upgrading Database Version from MySQL 5.7 to 8.0 Through DTS

Usage Instructions for TDSQL-C MySQL

New Version of Console

Implementing Multiple RO Groups with Multiple Database Proxy Connection Addresses

Strengths of Database Proxy

Selecting Billing Mode for Storage Space

Creating Remote Disaster Recovery by DTS

Creating VPC for Cluster

Data Rollback

Database/Table-Level Rollback

Overview

Database/Table Rollback by Time Point

Database/Table Rollback by Backup Set

Rolling back Entire Cluster

Overview

Solution to High CPU Utilization

How to Authorize Sub-Users to View Monitoring Data

Best Practices

Upgrading Database Version from MySQL 5.7 to 8.0 Through DTS

Last updated : 2024-06-07 14:21:10

Background

The database MySQL 8.0 of TDSQL-C for MySQL combines comprehensive management and control services with the new TXSQL kernel, ensuring faster, more stable enterprise-level services and a wide range of industry scenarios. Currently, the TDSQL-C for MySQL console does not support directly upgrading to MySQL 8.0. To upgrade, you can use the same cloud account to create a cluster with MySQL 8.0 in the source region, and then synchronize the data from the source cluster to the new cluster through Data Transfer Service (DTS). Finally, after confirming the data is accurate, delete the source cluster and switch the business to the new cluster to achieve the database version upgrade.

Use Cases

Synchronization Link	Source Database and Version	Target Database and Version
TDSQL-C for MySQL > TDSQL-C for MySQL	TDSQL-C for MySQL MySQL5.7 Synchronization between databases under the same cloud primary account	TDSQL-C for MySQL MySQL8.0

Notes

During full data synchronization, DTS consumes certain source instance resources, which may increase the load and pressure of the source instance. If your database configuration is low, we recommend you synchronize the data during off-peak hours.

To avoid duplicate data, ensure the tables to be synchronized have a primary key or a non-null unique key.

By default, a lock-free method is used during synchronization. Global locks (FTWRL) are not applied to the source database, only table locks are applied to tables without a primary key, and no locks are applied elsewhere.

During data synchronization, DTS will write a system database `__tencentdb__` in the source database using the account that executes the synchronization task, to record data comparison information during the synchronization

process.

To ensure that data comparison issues can be traced later, the synchronization task will not delete the `__tencentdb__` in the source database after completion.

The system database `__tencentdb__` occupies a very small space, roughly one thousandth to one ten-thousandth of the source database storage space (for example, if the source database's storage space is 50 GB, then the space occupied by the system database `__tencentdb__` will be approximately 5 MB to 50 MB). It also uses a single-thread, waiting connection mechanism, so it almost has no impact on the performance of the source database, nor does it pre-empt resources.

Prerequisites

The following permissions are required for the source database:



```
GRANT RELOAD, LOCK TABLES, REPLICATION CLIENT, REPLICATION SLAVE, SHOW VIEW, PROCES  
GRANT ALL PRIVILEGES ON `__tencentdb__`.* TO 'account'@'%';  
FLUSH PRIVILEGES;
```

The following permissions are required for the target database: ALTER, ALTER ROUTINE, CREATE, CREATE ROUTINE, CREATE TEMPORARY TABLES, CREATE USER, CREATE VIEW, DELETE, DROP, EVENT, EXECUTE, INDEX, INSERT, LOCK TABLES, PROCESS, REFERENCES, RELOAD, SELECT, SHOW DATABASES, SHOW VIEW, TRIGGER, and UPDATE.

Use Limits

Only synchronization of base tables, views, stored procedures, and functions is supported.

When synchronizing views, stored procedures, and functions, DTS will check whether the `DEFINER` corresponding to user1 ([DEFINER = user1]) in the source database and the synchronization account user2 are the same. If not, after the synchronization, DTS will modify the `SQL SECURITY` attribute of user1 in the target database from `DEFINER` to `INVOKER` ([INVOKER = user1]), while setting the `DEFINER` in the target database to the synchronization account user2 ([DEFINER = synchronization account user2]). If the definition of the view in the source database is too complex, the task may fail.

If the source is a non-GTID instance, DTS does not support source-end HA switching. Once the source instance switches, it may cause an interruption in DTS incremental synchronization.

Only synchronization of InnoDB, MyISAM, and TokuDB engine databases is supported. Tables using engines other than these three will be skipped by default. Among these, if the source database contains TokuDB engine data in compression mode, the target database must also support the compression mode for synchronization; otherwise, the task will report an error.

Interrelated data objects need to be synchronized together; otherwise, synchronization will fail. Common associations include: View Reference Table, View Referencing View, and Primary-Foreign Key Association Table.

During the incremental synchronization process, if the source database generates binlog statements in the `STATEMENT` format, it will cause the synchronization task to fail.

Gaps in the GTID of the source database binlog might affect the performance of the synchronization task and lead to task failure.

Scenarios that include both DML and DDL statements in one transaction are not supported. In such scenarios, an error may occur.

Geometry-related data types are not supported, and tasks will report an error when encountering such data types.

The `ALTER VIEW` statement is not supported; tasks encountering this statement will be skipped and not synchronized.

Operation Restrictions

Do not perform the following operations during synchronization, as they will cause the synchronization task to fail.

Do not modify or delete user information (including username, password, and permissions) and port number in both the source and target databases.

Do not write data with the binlog format as `STATEMENT` in the source database.

Do not clear binlogs on the source database.

During the incremental synchronization phase, do not delete the system database table `__tencentdb__`.

Supported SQL Operations for Synchronization

Operation Type	SQL Operation Statements
DML	INSERT, UPDATE, and DELETE
DDL	CREATE DATABASE, DROP DATABASE, ALTER DATABASE, CREATE TABLE, ALTER TABLE, DROP TABLE, TRUNCATE TABLE, RENAME TABLE, CREATE VIEW, DROP VIEW, CREATE INDEX, and DROP INDEX

Note:

The **CREATE TABLE table name AS SELECT** statement is not supported.

Environment Requirements

Type	Environment Requirements
Source Database Requirements	<p>The networks of the source and target databases must be interconnected.</p> <p>Instance Parameter Requirements:</p> <p>The row_format of the source database table cannot be set to FIXED.</p> <p>Binlog Parameter Requirements:</p> <p>The source binlog_format variable must be set to ROW.</p> <p>The source binlog_row_image variable must be set to FULL.</p> <p>Setting do_db and ignore_db is not allowed.</p> <p>We recommend you retain the binlogs of the source database for at least three days; otherwise, the task cannot be resumed from the checkpoint and will fail.</p> <p>Foreign Key Dependencies:</p> <p>Foreign key dependencies can only be set to two types: NO ACTION or RESTRICT.</p> <p>During partial database table synchronization, tables with foreign key dependencies must be completely migrated.</p> <p>The environment variable innodb_stats_on_metadata must be set to OFF.</p>
Target Database Requirements	<p>The version of the target database must be later than or equal to the version of the source database.</p> <p>The target database must have sufficient storage space. If the initialization type is set to "Full Data Initialization", the space of the target database must be at least 1.2 times the size of the source database's tables pending synchronization.</p> <p>The target database cannot have tables, views, or other synchronization objects with the same name as those in the source database.</p> <p>The max_allowed_packet parameter of the target database must be set to a minimum of 4M.</p>

Step 1: Creating a Synchronization Task

1. Access the [Data Sync Purchase Page](#), select the appropriate configuration, and click **Buy Now**.

Parameter	Description
Service Type	Select Data Sync.
Creation Mode	Select Create new task.
Billing Mode	Monthly subscription and Pay as you go billing modes are supported.
Source Instance Type	Select TDSQL-C for MySQL. It cannot be modified after purchase.
Source Instance Region	Select the region where the source instance is located. It cannot be modified after purchase.
Target Instance Type	Select TDSQL-C for MySQL. It cannot be modified after purchase.
Target Instance Region	Select the region where the target instance is located. It cannot be modified after purchase.
Specification	Select a specification based on your business needs. The higher the specification, the better the performance. For more details, refer to Billing Overview .

2. After the purchase, return to the [data synchronization list](#) to see the newly created data synchronization task. The newly created task must be configured before it can be executed.

3. In the data synchronization list, click **Configure** in the **Operation** column to enter the synchronization task configuration page.

Task ID / Name	Task Status / Progress	Running Mode	Specification	Billing Mode	Source Instance Type	Target Instance Type	Source Access Type
sync- sync-	<div></div> Status: Uninitialized	--	Small	Pay as you go	TDSQL-C MySQL	TDSQL-C MySQL	--

4. On the synchronization task configuration page, configure the source instance, account password, and target instance, account and password. After testing connectivity, click **Next**.

1 Set source and target databases >

2 Set sync options and objects >

3 Verify task

Task Configuration

Task Name *

Running Mode *

Immediate execution

Scheduled execution

Automatic Retry *

-

5

+

min

任务出错后，将在 5 分钟内按照预置重试策略进行重试

Source Database Settings

Source Instance Type *

TDSQL-C MySQL

Source Instance Region

North China(Beijing)

Access Type ⓘ *

Database

No Proxy

Cross-/Intra-Account *

Intra-account

Cross-account

Instance ID *

cynosdbmysql

Private IP: :3306

Account *

root

Password *

.....

Test Connectivity

✔ Test passed

Target Database Settings

Target Instance Type *

TDSQL-C MySQL

Target Instance Region

North China(Beijing)

Access Type ⓘ *

Database

No Proxy

Instance ID *

cynosdbmysql

Private IP: 3306

Account *

root

Password *

.....

Test Connectivity

✔ Test passed

Configuration Item	Parameter	Description
Task	Task	DTS will automatically generate a task name. You can set a name for the task as

©2013-2022 Tencent Cloud. All rights reserved.

Page 10 of 84

Configuration	Name	needed.
	Running Mode	Both immediate and scheduled execution modes are supported.
	Automatic Retry	After this feature is enabled, if the migration task is temporarily interrupted due to network anomalies or other reasons, DTS will automatically retry and resume the task within the set time frame, without the need for manual intervention by the user. The supported time range is 5 minutes to 720 minutes.
Source Database Settings	Source Instance Type	The source instance type selected at the time of purchase cannot be modified.
	Source Instance Region	The source instance region selected at the time of purchase cannot be modified.
	Access Type	Select TencentDB.
	Instance ID	Select the ID of the source instance.
	Account	Enter the account of the source instance. Account permissions must meet the requirements.
	Password	Enter the password of the source instance account.
Target Database Settings	Target Instance Type	The target instance type selected at the time of purchase cannot be modified.
	Target Instance Region	The target instance region selected at the time of purchase cannot be modified.
	Access Type	Select TencentDB.
	Instance ID	Select the ID of the target instance.
	Account	Enter the account of the target instance. Account permissions must meet the requirements.
	Password	Enter the password of the target instance account.

5. On the Set sync options and objects page, you can set the data initialization options, data synchronization options, and synchronization object options. After completing the settings, click **Save and Go Next**.

Note:

When **Initialization Type** is set to **Full data initialization** alone, the system assumes that table structures have already been created in the target database. Therefore, it will not synchronize table structures, nor will it verify if there are tables with duplicate names in the source and target databases. Therefore, if a user also selects **Precheck and report error** in the presence of **If Target Already Exists**, this validation and error-reporting feature will not be effective.

If you plan to use the rename operation on a table during synchronization (for example, renaming table A to table B), then the **Sync Object** must include the entire database (or the entire instance) where table A is located, not just table A. Otherwise, after the rename operation, data from table B will not be synchronized to the target database.

Data Initialization Option

Initialization Type

Structure initialization

Full data initialization

If Target Already Exists *

Precheck and report error

Ignore and execute

Data Sync Option

Primary Key Conflict Resolution *

Report

Ignore

Overwrite

The "Report" policy is only used in the incremental stage. To implement lock-free migration, the conflict resolution policy is

SQL Type

DML

Insert

Update

Delete

DDL

DDL

Custom DDL *

Sync Object Option

Sync Object ⓘ *

Specify object

Advanced Migration Object ⓘ

☐ Procedure

☐ Function

Advanced objects can only be copied once, which means you cannot copy new objects once the task is started.

Source Database Object ⓘ

Search library name, support fuzzy matching; click the drop-down to expand

ⓘ 1 database in total, with 1 displayed More

▶ ☒ mydatabase

Refresh Select all databases Clear

Selected Object ⓘ

Globally search for original

☒ mydatabas

Unfold all Fold all Select

▶ Advanced Options

Previous

Save and Go Next

Configuration Item	Parameter	Description
--------------------	-----------	-------------

Data Initialization Option	Initialization Type	<p>Structure initialization: During the execution of the synchronization task, the table structure from the source instance will be initialized in the target instance first.</p> <p>Full data initialization: During the execution of the synchronization task, data from the source instance will be initialized in the target instance first. In scenarios where only Full data initialization is selected, you need to pre-create the table structure in the target database.</p> <p>By default, both are selected, but can be canceled depending on the actual situation.</p>
	If Target Already Exists	<p>Precheck and report error: If a table with the same name exists, an error is reported and the process is not continued.</p> <p>Ignore and execute: Directly append both full and incremental data to the table in the target instance.</p>
Data Sync Option	Primary Key Conflict Resolution	<p>Report: If a primary key conflict is detected during synchronization, an error is reported and the data synchronization task is paused.</p> <p>Ignore: If a primary key conflict is detected during synchronization, the primary key record in the target database is retained.</p> <p>Overwrite: If a primary key conflict is detected during synchronization, the primary key record in the source database will overwrite that in the target database.</p>
	SQL Type	Supported operations are Insert, Update, Delete, and DDL. By selecting Custom DDL, you can choose different DDL synchronization policies as needed. For details, refer to Setting SQL Filter Policies .
Sync Object Option	Sync Object	Select specific objects, including base tables, views, stored procedures, and functions.
	Advanced Migration Object	Synchronization of advanced migration objects is a one-time action, only supporting synchronization of advanced objects that existed in the source database before the task started. After the task has started, newly added advanced objects will not be synchronized to the target database.

6. On the **Verify task** page, complete the verification and, after all verification items have passed, click **Start Task**.

Note:

If the verification fails, refer to [Solution to Verification Failure](#) to fix the issue and initiate the verification task again.

A synchronization task may increase the load on both the source and target database instances. Perform the operation during business off-peak hours.

Do not change the source or target database passwords after the synchronization task has started.

Do not execute DDL operations on the source database.

Do not disconnect the network or switch the source database.

Failed: Indicates that the verification item failed the check, the task is blocked, and you need to fix the issue and run the verification task again.

Alarm: Indicates that a verification item does not fully meet the requirements but the task can proceed, impacting the business to some extent. You need to decide based on the prompt whether to ignore the warning or fix the issue and continue.

Task ID / Name	Running Mode	Automatic Retry Time	Source Instance Type	Target Instance Type	Source Instance Access Type
sync-c sync-c	Immediate execution	5 分钟	TDSQL-C MySQL	TDSQL-C MySQL	Database

Initialization

If Target Already Exists

SQL Type

DML

DDL

Primary Key Conflict Resolution

Sync Object

Structure initialization/Full data initialization

Precheck and report error

Insert、Update、Delete

Select All

Report

[Unfold all](#) [Fold all](#)

Fuzzy search by source object name (case-insensitive)

1 database in total, with 1 found

mydatabase (Entire database selected)

Create Verification Task

Query Verification Result

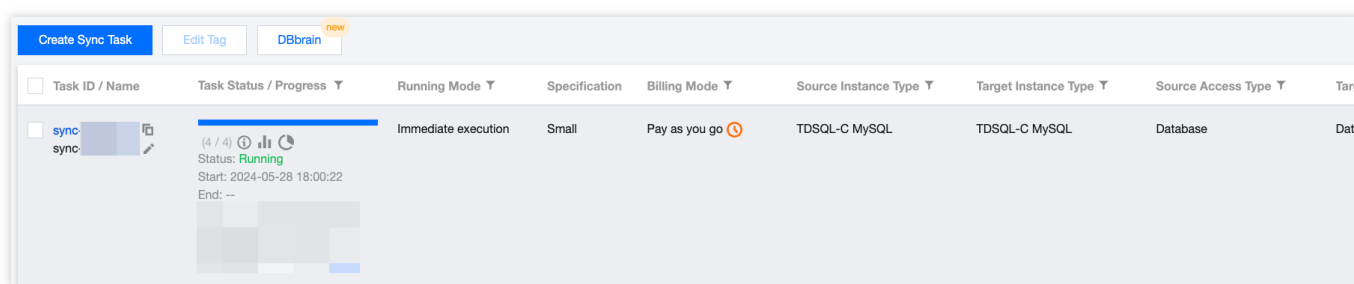
- connect db check
- necessary check
- version check
- source instance privilege check
- simple instance param check
- target instance privilege check
- check if target instance has conflict content
- check if there's enough space in target instance
- source instance binlog param check
- foreign key constraint check
- partial table foreign key constraint check
- view check
- warning param check

[Previous](#) [Verify Again](#) [Start Task](#)

7. Return to the data synchronization task list, and the task has entered the **Running** status.

Note:

In the **Operation** column, choose **More > Stop** to close a synchronization task. Ensure data synchronization is complete before closing the task.



Task ID / Name	Task Status / Progress	Running Mode	Specification	Billing Mode	Source Instance Type	Target Instance Type	Source Access Type	Target Instance Name
sync-1234567890	(4 / 4) Status: Running Start: 2024-05-28 18:00:22 End: --	Immediate execution	Small	Pay as you go	TDSQL-C MySQL	TDSQL-C MySQL	Database	Date

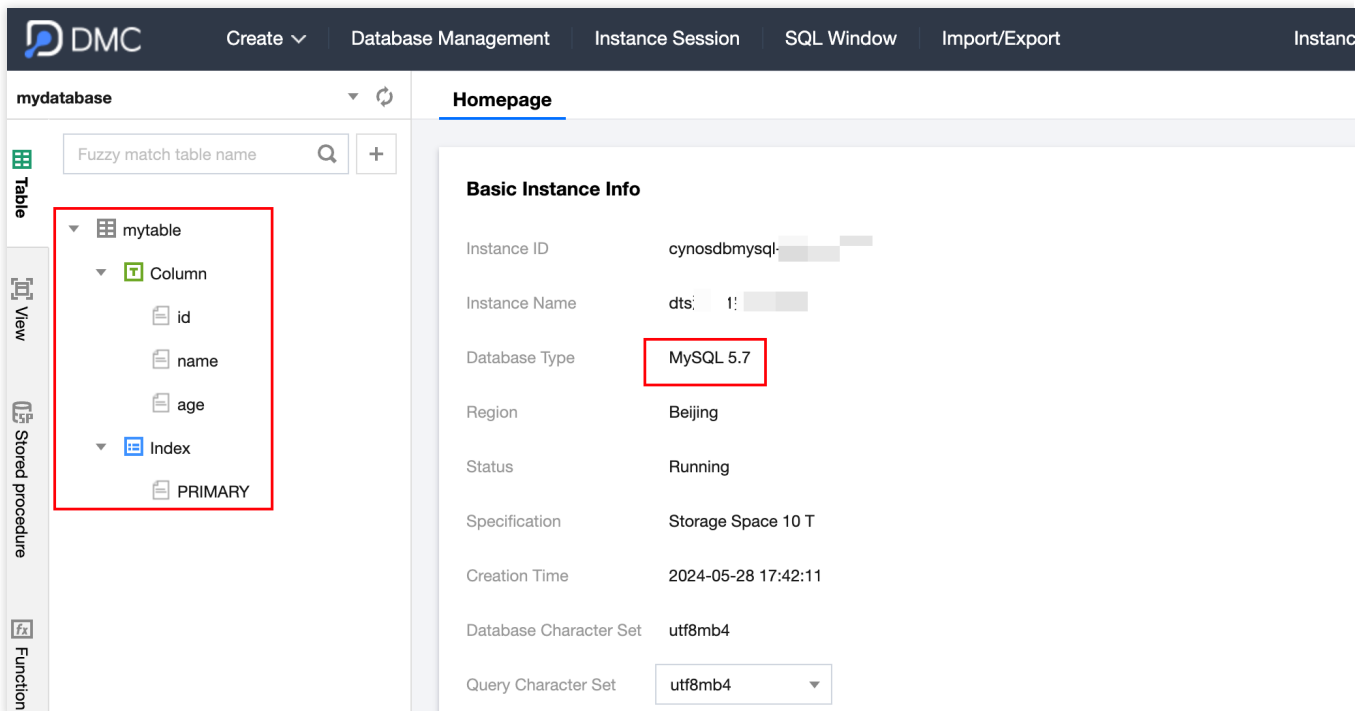
8. (Optional) You can click on the task name to enter the task details page and view the initialization state and monitoring data.

Step 2: Actively Verify Data Through DMC

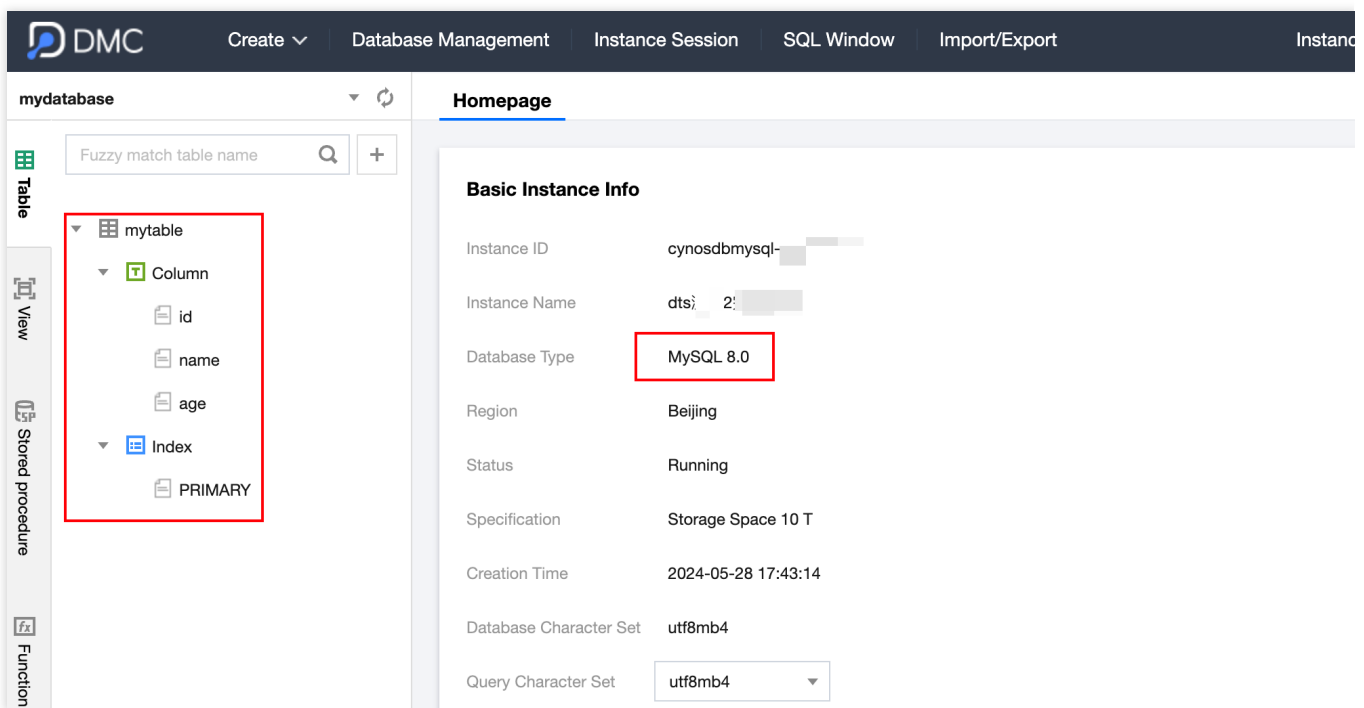
After the data synchronization task is created, data synchronization from the source cluster to the new cluster will continue. Considering the need to connect the business to a cluster with database version 8.0 in the future, refrain from writing incremental data to the source database to ensure consistency between the source database and new database.

1. Log in to the [TDSQL-C for MySQL console](#).
2. Find the source cluster used for data synchronization in the cluster list, and then in the top right corner of its cluster details page, click **Log In**.
3. In the pop-up window, enter your account password to access the source cluster's DMC management page.
4. Open a new webpage and log in to the [TDSQL-C for MySQL console](#).
5. Find the target cluster for data synchronization in the cluster list, and then in the top right corner of its cluster details page, click **Log In**.
6. In the pop-up window, enter the account and password to access the DMC management page of the target cluster.
7. On the DMC management page, compare the database tables and other data of the source cluster and the target cluster to ensure that all data has been fully synchronized.

Source Cluster DMC Management Page



Target Database DMC Management Page



Step 3: Switch the Business to the New Cluster

After the data synchronization is complete and consistent, and you confirm that the source cluster is no longer needed, you can manually update the business connection to the new cluster's connection address on the actual application side. Then, end the corresponding data synchronization task in the [data synchronization list](#) on the DTS side. Finally, delete the source cluster in the [TDSQL-C for MySQL console](#).

Stopping the synchronization task:

Create Sync TaskEdit TagDBbrainnew

Task ID / Name	Task Status / Progress	Running Mode	Specification	Billing Mode	Source Instance Type	Target Instance Type	Source Access Type	Targ
<div>sync</div> <div>sync</div>	<div>(4 / 4)</div> <div>Status: Running</div> <div>Start: 2024-05-28 18:00:22</div> <div>End: --</div>	Immediate execution	Small	Pay as you go	TDSQL-C MySQL	TDSQL-C MySQL	Database	Date

Note:

Before the service ends, ensure there are no business dependencies on this synchronization service.

Existing configured synchronization tasks will be stopped.

The synchronization task will be set to stopped status.

For monthly subscription data synchronization tasks, stopping them will not automatically trigger a refund. For refunds, click **Return/Refund**.

Deleting the source cluster:

<< Hide cluster lists (5)CreateQuick CheckRenewMore

● dts-2

cynosdbmysql-m

Beijing Zone 3 | MySQL 8.0

● dts-1

cynosdbmysql

Beijing Zone 3 | MySQL 5.7

● cynosdbmysql-

cynosdbmysql-f

Beijing Zone 3 | MySQL 5.7

● cynosdbmysql

cynosdbmysql

dts-1

ID cynosdbmysql-t

Running

Database Version 2.1.12

Details/Upgrade

Billing Mode Compute: Pay as You Go/Storage

Project Default Project

Adjust

Transfer Linkage High IO

Tag

Billing Mode Compute: Pay as You Go/Storage

Deployment Mode Single-AZ

Compatible Database MySQL 5.7

Cluster Details

Monitoring and Alarms

Account Management

Database Management

Database Proxy

Parameter Settings

Security Group

Usage Instructions for TDSQL-C MySQL

Last updated : 2024-06-07 14:22:07

Purpose

Standardize the management and maintenance of TDSQL-C for MySQL to avoid unavailability and other issues caused by improper operations.

Guide database developers in writing SQL statements reasonably to make best use of performance of TDSQL-C for MySQL.

Permission Management Specifications

Considering the stability and security of TDSQL-C for MySQL, SUPER, SHUTDOWN, and FILE permissions are restricted. Therefore, the following error may occur when executing SET statements on TDSQL-C for MySQL:



```
#1227-Access denied;you need(at least one of)the SUPER privilege (s) for this opera
```

Solution: If you need to modify related parameters using SET, this can be done through the **Parameter Settings** feature on the cluster management page of the console.

Grant permissions on demand. It is sufficient to grant general applications only with the DML permissions (SELECT, UPDATE, INSERT, and DELETE).

Grant permissions to users of general applications at the database level, following the principle of least privilege.

Allow authorized users to access TDSQL-C for MySQL only from specific IP addresses or IP ranges. This can be achieved by configuring security groups in the console as instructed there. To set a security group for public network

access, be sure to allow all the egress IP addresses involved.

Use different accounts for management and development.

Operation Specifications

Notes

Do not use weak passwords for enhanced database instance security.

For private network connections and login, ensure that the client-side CVM and TDSQL-C for MySQL instances are on machines with the same account, in the same region, and on the same VPC.

To parse binlog logs downloaded from the console locally, ensure the client's MySQL version matches the TDSQL-C for MySQL cluster version. Otherwise, parsing may result in garbled text. We recommend that you use `mysqlbinlog` v3.4 or later.

Enclose the URL with quotation marks when downloading cold backup files to a CVM instance over the private network in the console; otherwise, a 404 error will occur.

Suggestions

Avoid performing online DDL operations during peak business hours. For applicable tools, refer to: `pt-online-schema-change` .

Avoid performing batch operations during peak hours. Do such operations in off-peak hours instead.

Avoid running an instance for multiple businesses to minimize the risk of mutual interference between businesses due to high coupling.

We recommend that you disable transaction auto-commit and adopt the habit of using `begin;` first in online operations to reduce the risk of data loss from faulty operations. In case of faulty operations, you can use the rollback feature of TDSQL-C for MySQL. For tables not involving cross-database and cross-table logic, rapid or instant rollback can be used for faster data restoration. The default name for newly generated tables upon rollback is

`original table name_bak` .

Estimate the resources required in advance and optimize the instances for promotional campaigns of your business. In case of a great demand for resources, contact your service manager in a timely manner.

Database and Table Design Specifications

Notes

TDSQL-C for MySQL v5.7 and later do not support MyISAM or MEMORY engines. If the MEMORY engine is required, we recommend that you use TencentDB for Redis or Memcached. If databases are migrated to TDSQL-C for MySQL via DTS, the MyISAM engine will be automatically converted to InnoDB.

Create at least one index on the auto-increment column or create a composite index whose first column is the auto-increment column.

`row_format` must be non-fixed.

Every table must have a primary key. Even if no suitable column can serve as the primary key, it is still necessary to add an arbitrary column as the primary key. According to the First Normal Form standard of MySQL, InnoDB secondary index leaf nodes will store a copy of the primary key value. We recommend that you use an auto-incrementing short column as the primary key to reduce the disk space occupied by indexes and improve efficiency. In scenarios where `binlog_format` is set to row, deleting data in batches without a primary key can cause severe primary-secondary latency.

Define fields as NOT NULL and set default values. NULL fields will cause unavailability of indexes, thus bringing problems to SQL development. NULL calculation can only be implemented based on IS NULL and IS NOT NULL.

Suggestions

Plan database resource usage based on business scenario analysis and estimated data access metrics (including database read/write QPS, TPS, storage and space). You can also configure monitoring for TDSQL-C for MySQL instances through the console in the Tencent Cloud Observability Platform (TCOP) interface.

Put the tables for the same type of businesses into one database when building databases and try not to mix them. Do not perform cross-database correlation operations in programs, as doing so will affect subsequent quick rollbacks.

Always use the utf8mb4 character set to reduce the risk of garbled characters. Certain complex Chinese characters and emojis can be displayed correctly only using utf8mb4. Changing the character set only affects tables created after the change. Therefore, we recommend that you select utf8mb4 when purchasing a new TDSQL-C for MySQL instance.

Use the DECIMAL type to store decimal values. The FLOAT and DOUBLE types have insufficient precision, especially for businesses involving money where the DECIMAL type must be used.

Do not use the TEXT or BLOB type to store a large quantity of text, binary data, images, files, and other contents in a database; instead, store such data as local disk files and only store their index information in the database.

Avoid using foreign keys. We recommend that you implement the foreign key logic at the application layer. Foreign key and cascade update are not suitable for high-concurrency scenarios, because they may reduce the insertion performance and lead to deadlock in case of high concurrency.

Reduce the coupling between business logic and data storage, mainly use databases for data storage, and aim to implement business logic at the application layer as much as possible, minimizing the use of stored procedures, triggers, functions, events, views, and other advanced features due to their poor portability and scalability. If such objects exist in an instance, we recommend that you not set definer by default to avoid migration failures caused by inconsistencies between the migration account and definer.

Do not use partitioned tables if your business is not expected to reach a significant magnitude in the short term. Partitioned tables are primarily used for archive management, especially in the express industry and e-commerce industry for order tables. Partitioned tables do not enhance performance, unless over 80% of queries in your business involve the partitioning field.

Purchase read-only instances to implement read/write separation for business scenarios with a high read load and low requirement for consistency (where a data latency within seconds is acceptable).

Index Design Specifications

Notes

Do not create indexes on columns that are updated very frequently and have low discrimination. Record updates will modify the B+ tree, and indexing fields with frequent updates can significantly reduce the database performance. When creating a composite index, place the column with the highest discrimination on the far left. For example, in `select xxx where a = x and b = x;`, if a and b are used together to create a composite index and a has higher discrimination, then create `idx_ab(a,b)`. If you have a mix of Non-Equal and Equal conditions, the column with the Equal condition must be placed first. For instance, in `where a xxx and b = xxx`, b must be placed in the foremost position of the index, even if a has higher discrimination, as the index cannot reach a.

Suggestions

Use no more than five indexes in a single table and no more than five fields in a single index. Too many indexes may affect the filtering effect, occupy much more capacity, and consume more resources for management.

Create indexes on the columns that are used for SQL filtering most frequently with a high cardinality value. It is meaningless to create indexes on a column not involved in SQL filtering. The higher the uniqueness of a field, the higher the cardinality value, and the better the index filtering result. Generally, an index column with a cardinality below 10% is considered an inefficient index, such as the gender field.

When indexing a VARCHAR field, we recommend that you specify the index length, rather than indexing the entire column. Generally, VARCHAR columns are long, and specifying a certain index length is often sufficient for discrimination. It is not necessary to index the entire column, which can be cumbersome and increase the cost of index maintenance. The index discrimination can be evaluated using `count(distinct left(column name, index length))/count(*)`.

Avoid using redundant indexes. If both index (a,b) and index (a) exist, (a) is considered a redundant index. If the query filtering is based on column a, the index (a,b) is sufficient.

Reasonably utilize covering indexes to reduce the I/O overhead. In InnoDB, leaf nodes of secondary indexes only save their own key values and primary key values. If an SQL query does not target an index column or primary key, the search through this index will first find the corresponding primary key and then locate the required column based on that primary key, leading to back table. This process incurs additional I/O overhead. Covering indexes can be employed to solve this issue. For example, `select a,b from xxx where a = xxx`, if a is not the primary key, then a composite index can be created for the columns a and b, thereby avoiding back table.

SQL Statement Compiling Specifications

Notes

Do not use LIMIT for UPDATE and DELETE operations, because LIMIT is random and may cause data errors; instead, you must use WHERE for such operations for exact match.

Do not use `INSERT INTO t_XXX VALUES (xxx)`. You must explicitly specify the column attributes to be inserted to prevent data errors due to changes in the table structure.

The following are common causes of invalid indexes in SQL statements:

Implicit type conversion, such as when the type of index a is VARCHAR and the SQL statement is written as where a = 1; where VARCHAR is converted to INT.

Math calculations and functions are performed on the index columns; for example, the date column is formatted using a function.

The character set of join columns is not unified.

Inconsistencies in multi-column sorting order, such as when the index is (a,b), but the SQL statement is ordered by a b desclike.

When fuzzy search is used, indexes can be queried for character types in the format of `xxx%`; other scenarios will not use indexes.

Negative direction queries are used (such as not, !=, and not in,).

Suggestions

Ensure query on demand and reject `select *` to avoid the following issues:

The covering index does not work and the problem of TABLE ACCESS BY INDEX ROWID occurs, which leads to extra I/O overhead.

Additional memory load occurs. A large amount of cold data is imported to `innodb_buffer_pool_size`, which may reduce the query hit rate.

There are extra overheads in network transfer.

Avoid using large transactions. We recommend that you split a large transaction into multiple small ones to avoid primary-secondary latency.

Unnecessary lock waits may occur. To prevent this issue, commit transactions in the business code timely.

Minimize the use of join operations for multiple tables and do not perform join operations on big tables. When a join operation is performed on two tables, the smaller one must be used as the driving table, the columns to be joined must have the same character set, and all of them must have been indexed.

Optimize pagination with LIMIT, operations like LIMIT 80000,10 extract 80010 records and return the last 10, creating significant database stress. We recommend that you determine the starting record position before paginating, for example, `SELECT * FROM test WHERE id >= (SELECT sql_no_cache id FROM test order by id LIMIT 80000,1) LIMIT 10;`

Avoid using an SQL statement with multi-level nested subqueries. The query optimizer prior to MySQL v5.5 can convert IN to EXISTS and does not go through the indexes. In this case, a large external table may result in poor performance.

Note:

It is difficult to completely avoid the aforementioned issues. The solution is to set the aforementioned conditions as secondary filtering conditions for indexes rather than as primary filtering conditions.

A large number of full table scans have been detected through monitoring. You can set the

`log_queries_not_using_indexes` parameter in the console and later download the slow log file for analysis.

However, keep it enabled for only a short duration to avoid a surge in slow logs.

Perform the required SQL audit before a business is released. In routine operation and maintenance work, download slow query logs regularly for targeted optimization.

New Version of Console

Last updated : 2023-11-09 11:21:59

TDSQL-C for MySQL has released a new version of the console to support tab views. Compared with the list view of the old version, the new version offers improved usability, visualization, and efficiency. We recommend that you use the new version of the console as it provides a clearer view of cluster information and allows for more convenient and efficient cluster management.

This document describes the new version of the console, including a comparison between the old and new version as well as the feature management in the new version.

Comparison between new and old versions of the console

UI comparison

New version

Old version

The new version of the console adopts a tab view, placing more emphasis on displaying information at the cluster level. The cluster architecture and deployment are presented in a clearer manner. The toggle switch between tab view and list view is retained.

TDSQL-C Cluster List Guangzhou 6 Other regions 9 Tab View

[Hide cluster lists \(6\)](#) [Create](#) [Quick Check](#) [Renew](#) [More](#) Separate keywords with "|"; press Enter to

cynosdbmysql-
cynosdbmysql-
Guangzhou
Zone 4
MySQL
5.7

cynosdbmysql-
cynosdbmysql-
Guangzhou
Zone 4
MySQL
5.7

cynosdbmysql-
cynosdbmysql-
Guangzhou
Zone 4
MySQL
5.7

cynosdbmysql-
cynosdbmysql-
Guangzhou
Zone 6
MySQL
5.7

cynosdbmysql-
cynosdbmysql-
Guangzhou
Zone 6
MySQL
5.7

cynosdbmysql-
cynosdbmysql-
Guangzhou
Zone 4
MySQL
5.7

cynosdbmysql-6 Running [Log](#)

ID cynosdbmysql-

Database Version **2.1.10** [Upgrade](#)

Project **Default Project** [Adjust](#)

Transfer Linkage **High IO**

Tag

Billing Mode **Compute: Pay as You Go/Storage: F**

Deployment Mode **Multi-AZ**

[Cluster Details](#) [Monitoring and Alarms](#) [Account Management](#) [Database Management](#) [Database Pro:](#)

Guangzhou Zone 4

[Enable Database Proxy](#)

The old version of the console adopts a list view, where information is displayed horizontally in the list. However, this layout is affected by screen scaling and can easily result in folded or truncated information.

TDSQL-C Cluster List Guangzhou 6 Other regions 9

[Create](#) [Quick Check](#) [Renew](#) [More](#) Separate keywords with "|"; press Enter to se

<input type="checkbox"/>	Cluster ID/Name	Cluster ...	Compatible Dat...	AZ	Read/Write Address	Read-Only Address	Database Proc
<input type="checkbox"/>	cynosdbmysql- cynosdbmysql-	Running	MySQL 5.7	Guangz...	(Private) :4:3306 (Public) Disabled	--	--
<input type="checkbox"/>	cynosdbmysql- cynosdbmysql-	Paused	MySQL 5.7	Guangz...	(Private) :6:3306 (Public) Disabled	--	--

Advantage comparison

Comparison Item	New version	Old version	Description
Ease of use	☆☆☆☆	☆☆	The new version of the console reduces the steps required to access the management page by eliminating the need to go back to the cluster list. You can smoothly switch between clusters and view their basic information without having to return to the cluster list.

			<p>The new version of the console provides a more user-friendly approach to updating cluster information. For example, on the details page, you can easily manage and modify the configuration, address, port, expiration time of read-write instances or read-only instances. Additionally, you can quickly add read-only instances, enable database proxy, and restart instances directly from the details page. The new version of the console has optimized the display of prompts, improved the presentation of the source and replica AZs for cluster status, and enhanced the display style of expiration time. These optimizations make it easier for users to intuitively obtain information.</p>
Readability	★★★★	★★	<p>The new version of the console makes it easier to read the cluster storage information on the cluster details page. The storage mode and storage space usage are presented clearly, making it more convenient for managing the storage space.</p> <p>The new version of the console displays the cluster deployment architecture diagram on the cluster details page, including the database proxy, read-write instances, and read-only instances. This diagram provides a clear representation of the relationships between the deployed components within the cluster.</p>
Logicity	★★★★	★★	<p>The new version of the console has enhanced logicity in the presentation and usage of clusters. You can first access basic information and the lifecycle of the corresponding cluster on the cluster management page. Then, on the cluster details section, you can see the relationships between read-write instances, read-only instances, and storage space, that is, a cluster can have one read-write instance and up to 15 read-only instances, and they all share the same storage.</p>

Feature management in the new version of the console

The new version of the console has optimized and added some management operations. Below are the optimized and newly added features, along with their specific operation options.

Overview

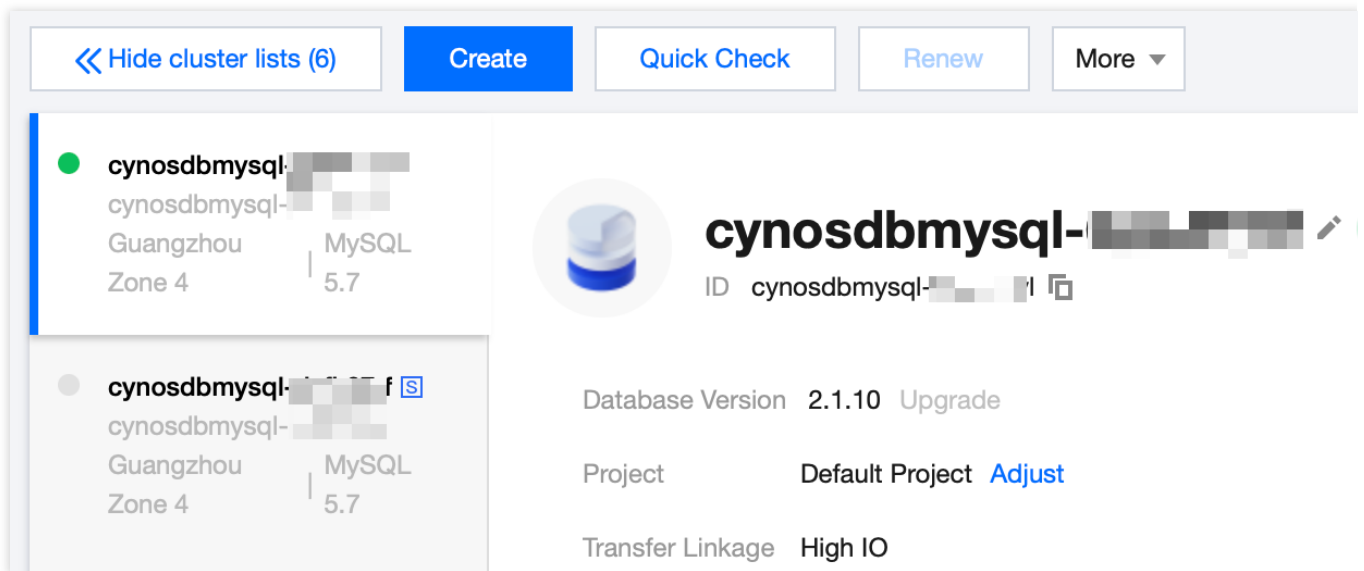
Location	Operation
Cluster list	Hide cluster list

	Expand cluster list
Cluster management	Modify cluster name
	Upgrade database kernel version
	Adjust instance project
	Edit tag
	Set renewal
	Manage renewal
	Modify storage billing mode
	Modify deployment mode
	Log in to a cluster
	Terminate a cluster
	Change from pay-as-you-go to monthly subscription
	Change from pay-as-you-go to serverless
Cluster details	Perform the source-replica switch
	Enable database proxy
	Restart an instance
	Enter the instance details page
	Modify instance name
	Adjust instance configuration
	Change instance network
	Modify instance connection address
	Enable public network
	Terminate an instance

Hide/expand cluster list

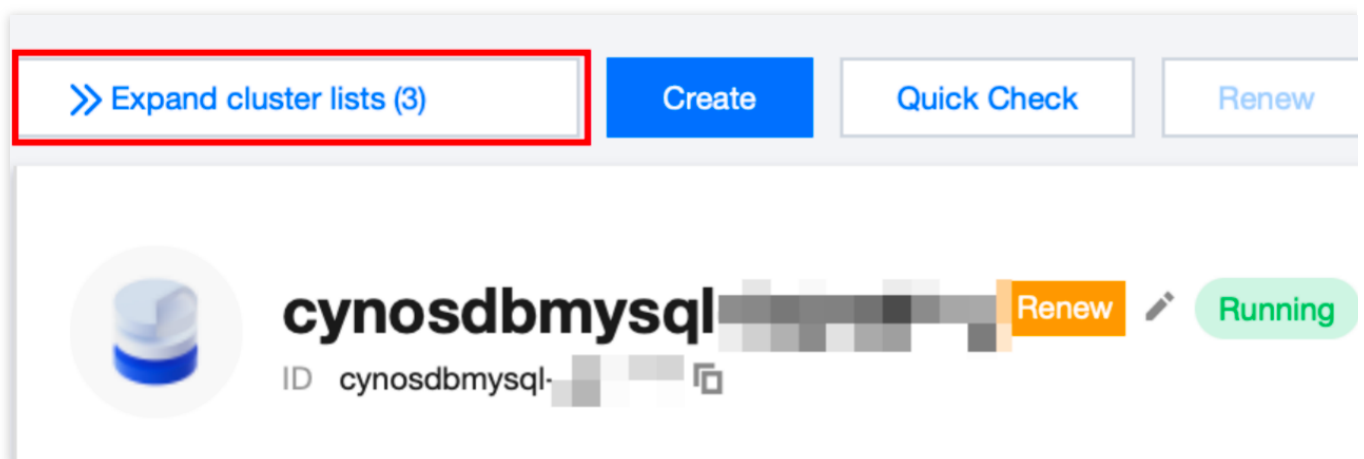
With this feature, you can hide other clusters and only display the information of the currently focused cluster. You can also expand the cluster list for easy switching to other clusters.

Hide cluster list:



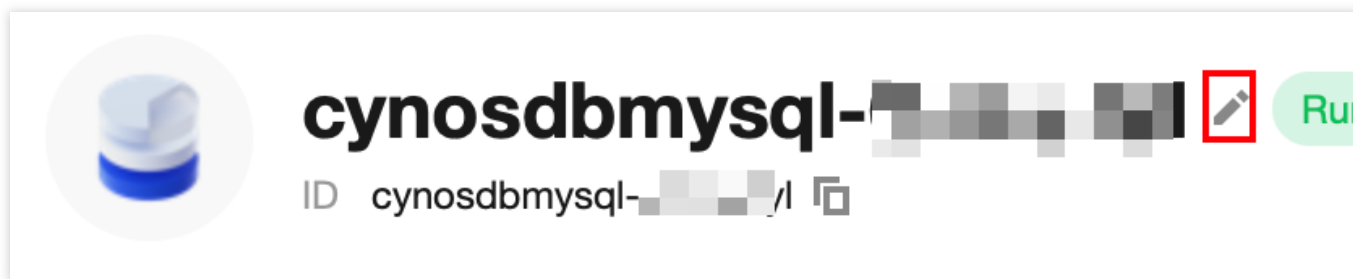
Expand cluster list

:



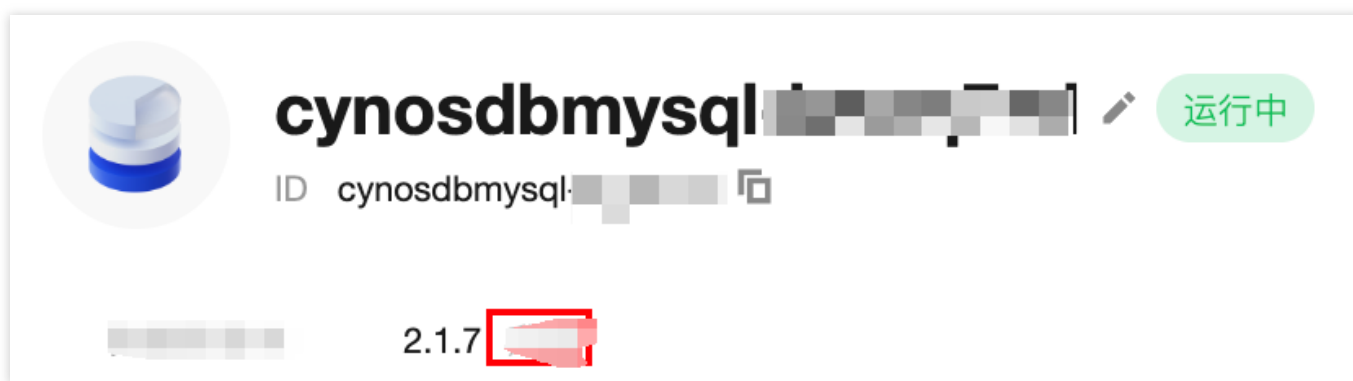
Modify cluster name

On the cluster management page, click the edit icon next to the cluster name.



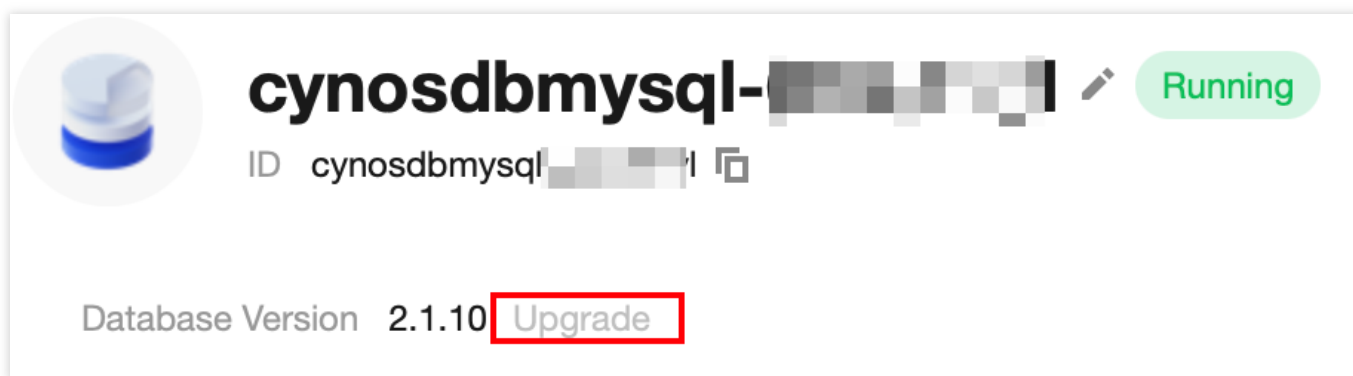
Upgrade database kernel version

On the cluster management page, click **Upgrade** next to the database version.



Adjust instance project

On the cluster management page, click **Adjust** next to the project



Edit tag

On the cluster management page, click the edit icon next to the tag.

Transfer Linkage High IO

Tag



Set renewal

If the billing mode for the cluster compute nodes is monthly subscription, the cluster can be renewed. On the cluster management page, after configuring the renewal settings, click **Set** next to **Renewal Settings**.

Project

Default Project [Adjust](#)

Renewal Settings [Set](#)

Manage renewal

If the billing mode for the cluster compute nodes is monthly subscription, the cluster renewal can be managed (renew, set to manual renewal, set unified expiration date, enable non-renewal). On the cluster management page, after the cluster has reached its expiration date, click on **Renewal Management**.

Cluster Expiration Time 2023-12-04 10:26:30 [Renewal Manage](#)

Deployment Mode Single-AZ

Modify storage billing mode

If compute nodes in a cluster are pay-as-you-go, its storage nodes are likewise billed in this way. But if the compute nodes are monthly subscribed, the storage billing mode can be changed from pay-as-you-go to monthly subscription. You can go to the **Cluster Management** page > **Billing Mode** > Storage, and then click on the edit icon next to **Storage**.

Billing Mode Compute: Monthly Subscription/Storage: Monthly Subs

Cluster Expiration Time 2024-01-29 10:53:47

Deployment Mode Single-AZ 

Modify deployment mode

On the cluster management page, click the edit icon next to **Deployment Mode**.

Deployment Mode Single-AZ 

Log in to a cluster

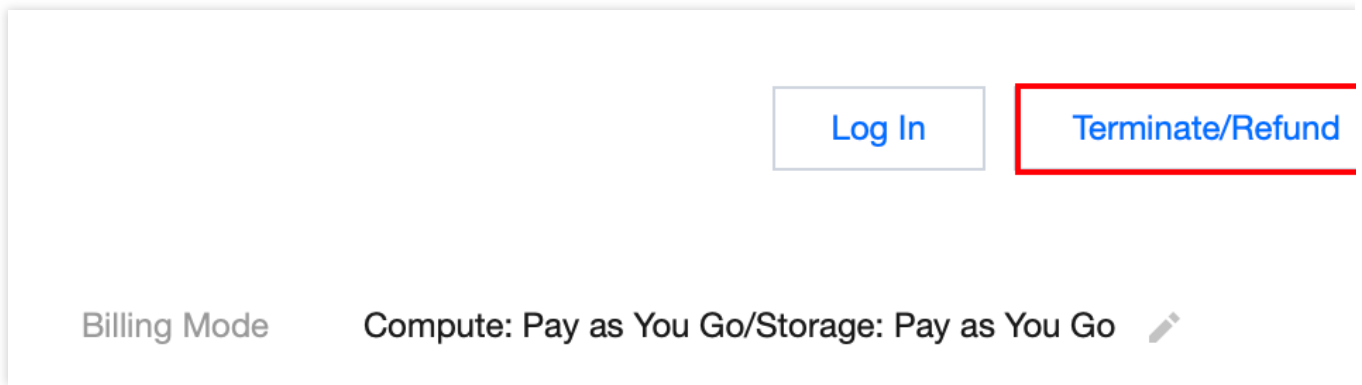
On the cluster management page, click **Log In**.

Log In

Terminate/Refund

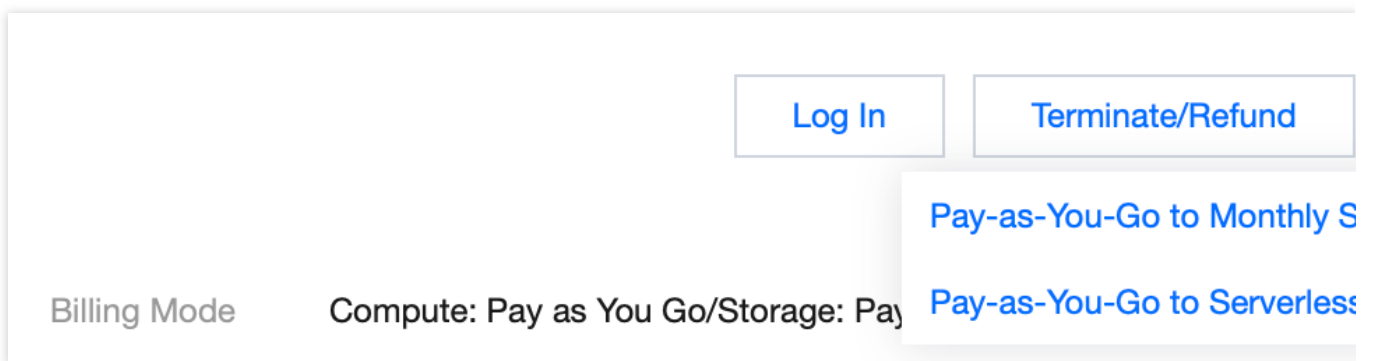
Terminate a cluster

On the cluster management page, click **Terminate/Refund**.



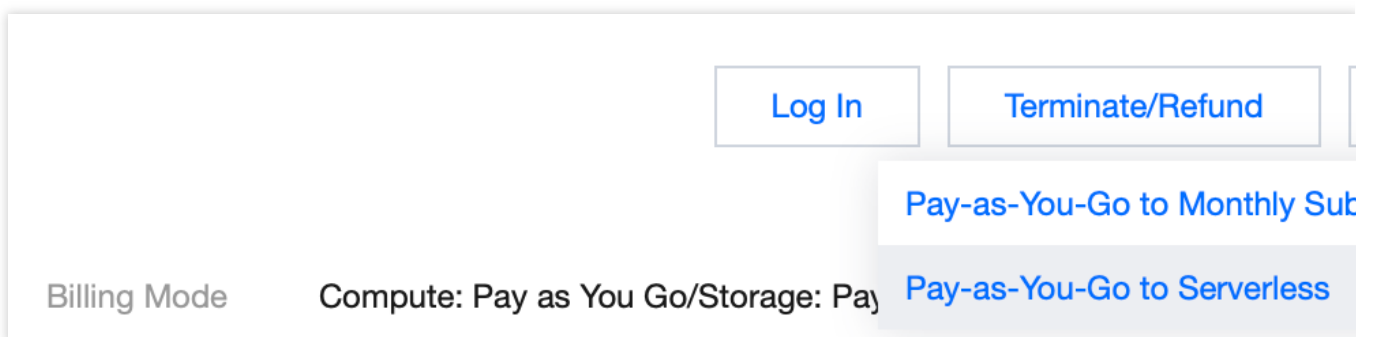
Change from pay-as-you-go to monthly subscription for the cluster computing nodes

On the cluster management page, click **More > Pay-as-You-Go to Monthly Subscription**.



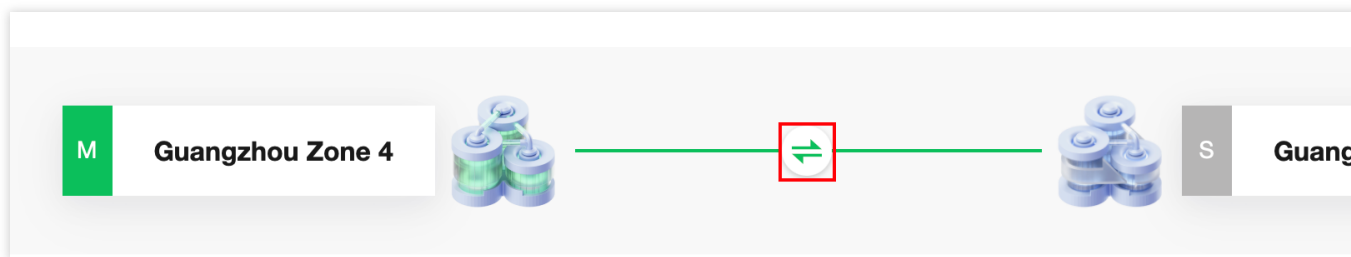
Change from pay-as-you-go to serverless for the cluster computing nodes

When the cluster compute nodes are monthly subscribed, their billing mode can't be changed to other options. If the cluster is deployed across multiple AZs, the compute nodes doesn't support switching from pay-as-you-go to serverless in billing mode. On the cluster management page, click **More > Pay-as-You-Go to Serverless**.



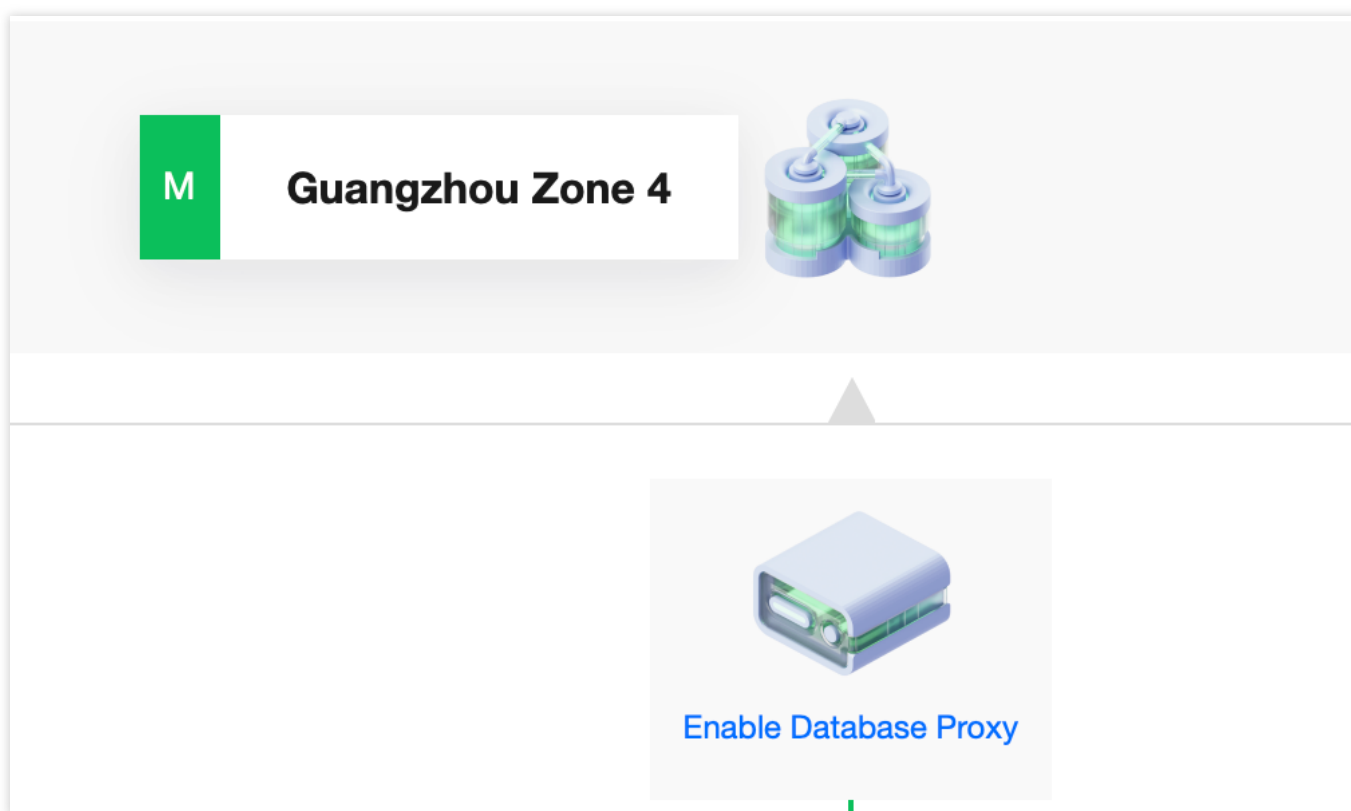
Perform the source-replica switch

On the architecture diagram of the cluster details page, click the switch button to perform the source-replica switch.



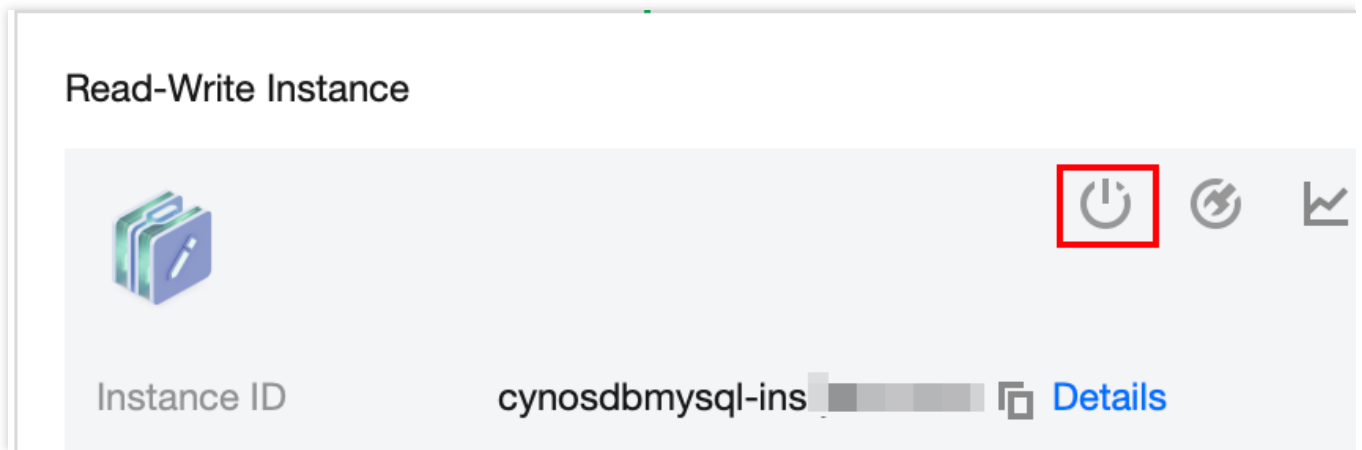
Enable database proxy

On the cluster details page, click **Enable Database Proxy**.



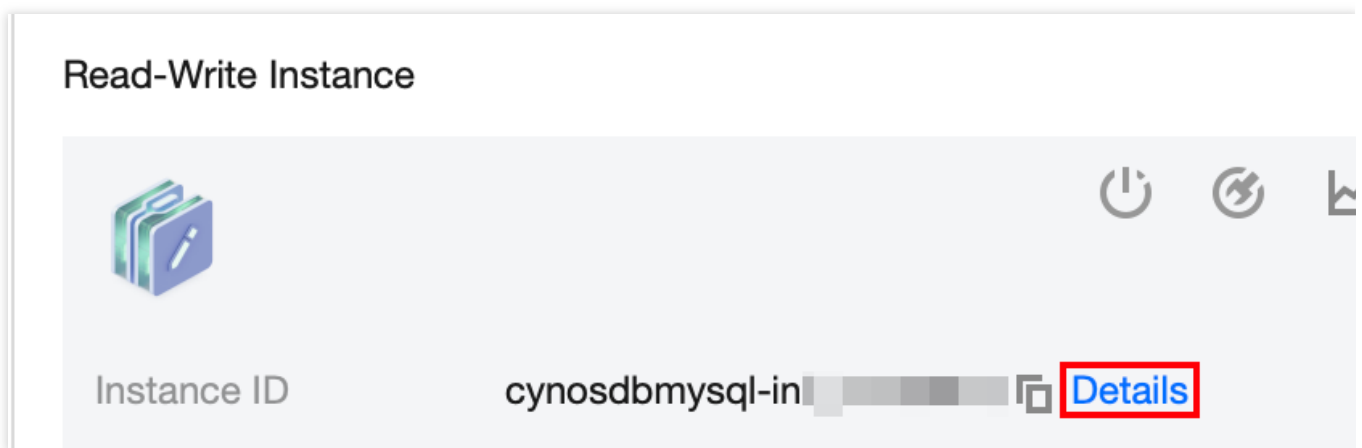
Restart an instance

On the cluster details page, find the target instance and click the restart icon.



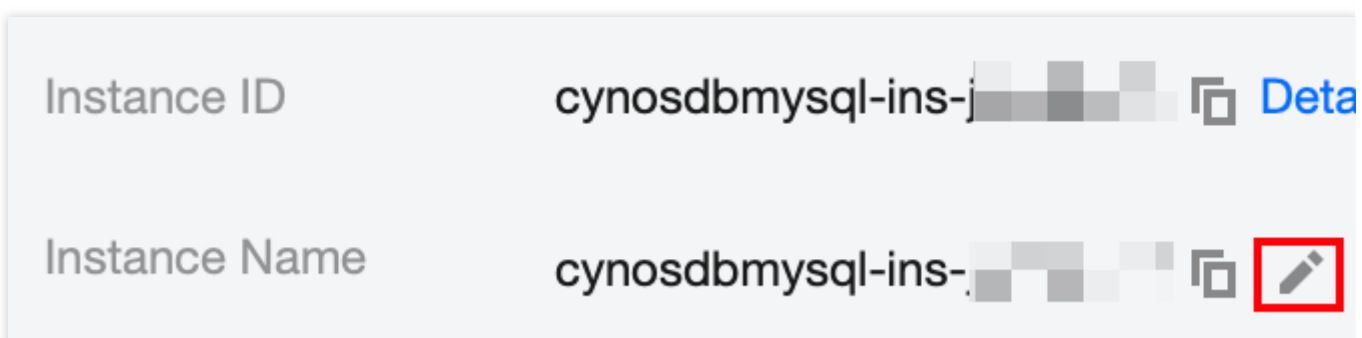
Enter the instance details page

On the cluster details page, find the target instance and click **Details**.



Modify instance name


On the cluster details page, find the target instance and click the edit icon next to the instance name.







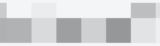





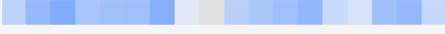



Adjust instance configuration

On the cluster details page, find the target instance and click the configuration adjustment icon, or click **Adjust Configurations** next to the configuration.

Read-Write Instance






Instance ID	cynosdbmysql-in-   Details
Instance Name	cynosdbmysql-ins-   
Configuration	1-core/2 GB MEM Adjust Configurations
Status	● Running 
Instance Type	Read-Write Instance
Network	 Change Network
Read/Write Address	<div><div>Private</div>Host 6.31 Port 3306 </div> <div><div>Public</div>-- Enable</div>

Change instance network

On the cluster details page, find the target instance and click **Change Network** next to the network it belongs to.

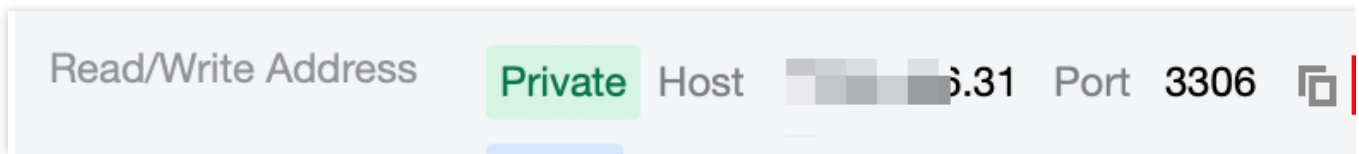
Instance Type

Read-Write Instance

Network	 Change Network
---------	---

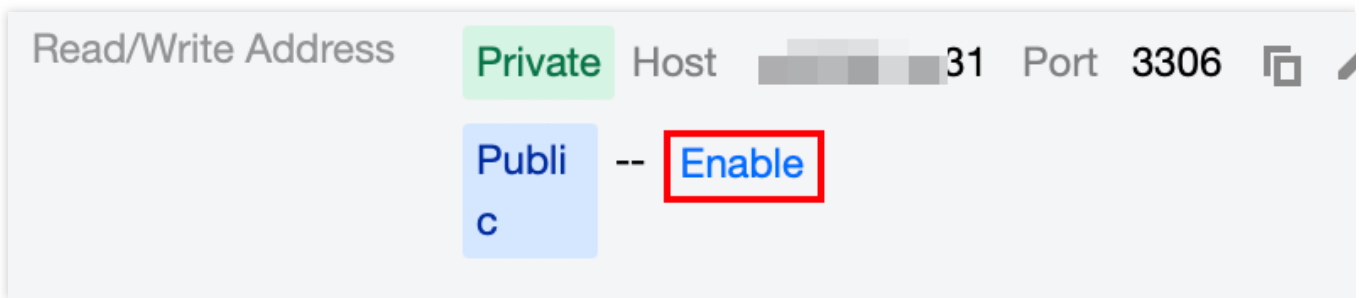
Modify instance connection address

On the cluster details page, find the target instance and click the edit icon next to the read-write address or read-only address.



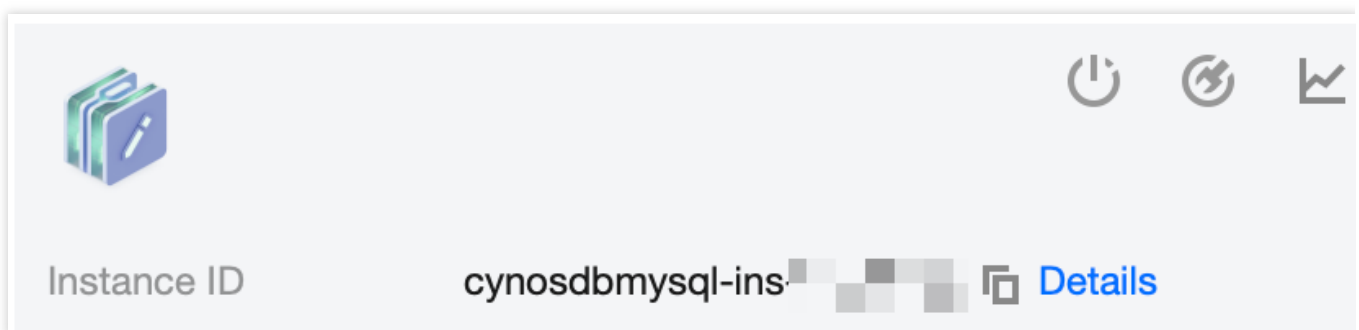
Enable public network

On the cluster details page, find the target instance and click **Enable** next to the public network of the read-write address or read-only address.



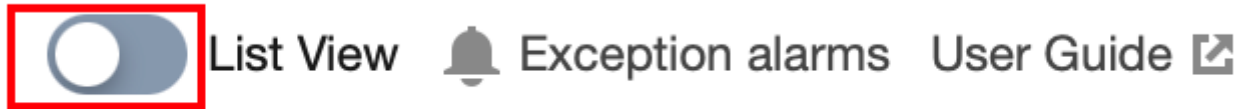
Terminate an instance

If there are read-only instances in the cluster and you need to delete the read-write instance, you need to first delete all the read-only instances in the cluster before you can delete the read-write instance. On the cluster details page, find the target instance and click the termination icon.

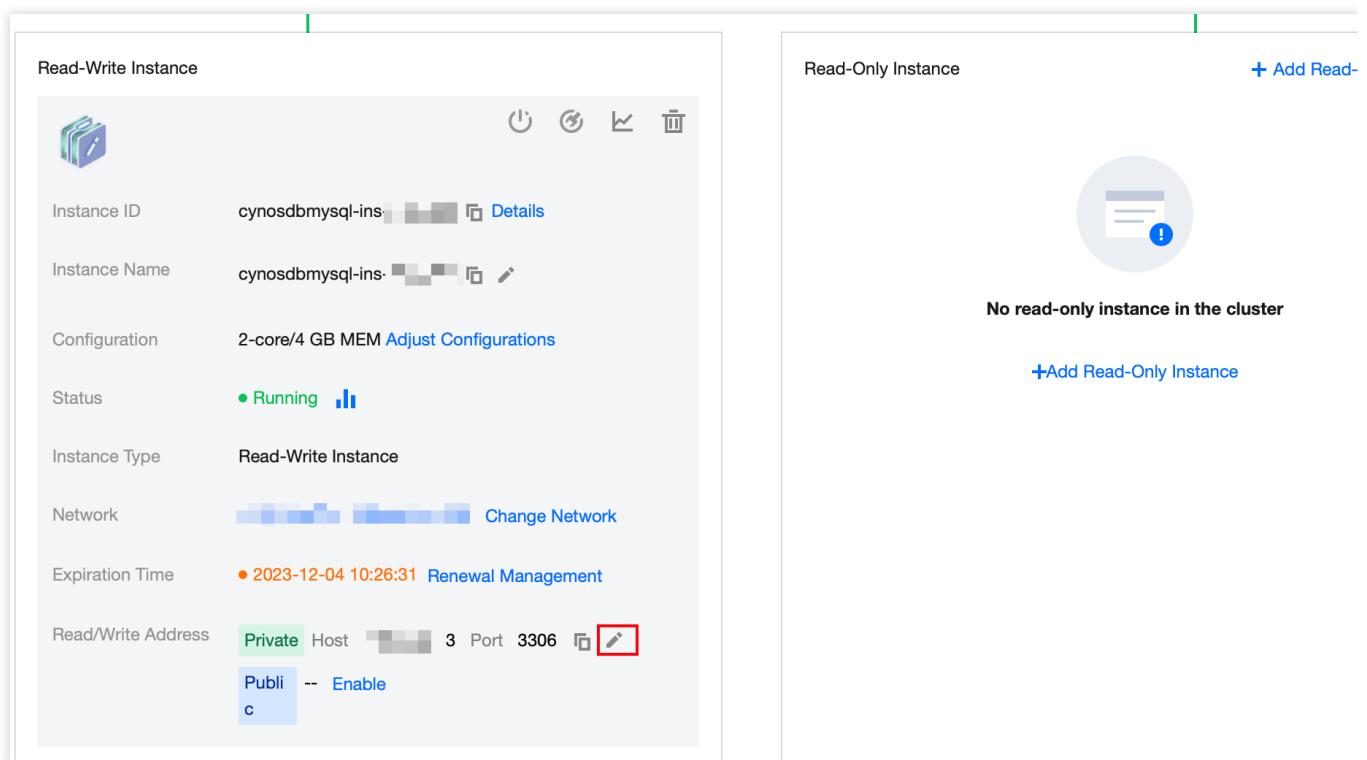


Example 1. Modify the instance connection address in the new version of the console

1. Log in to the [TDSQL-C for MySQL console](#).
2. After login, if the cluster list page is in list view, you can toggle on the button in the top right corner to enable the console view mode, and switch from list view to tab view.



3. Find the target cluster in the cluster list and click it to enter the cluster management page.
4. Find the target instance (read-write instance or read-only instance) on **Cluster Management > Cluster Details**, and click the edit icon next to its read-write address or read-only address.



5. In the pop-up window, set the new private address and port, set **Valid Hours of Old IPs**, and click **OK**.

Example 2. Adjust instance configuration in the new version of the console

Adjusting compute nodes

If the compute nodes are overprovisioned or underprovisioned, you can adjust their specifications to fully utilize resources and reduce unnecessary costs.

For detailed directions, see [Adjusting Compute Configuration](#).

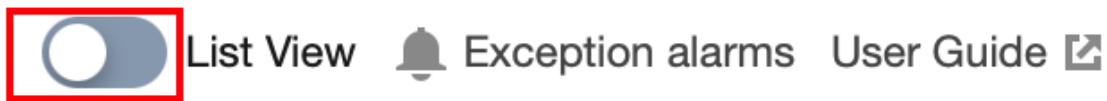
Adjusting storage space

If the cluster storage space cannot meet your business needs, you can adjust it as needed.

The storage space will be adjusted in the following three scenarios: (1) when the storage billing mode is pay-as-you-go and you need to use more storage space than the current compute specification limit; (2) when the storage billing mode is pay-as-you-go and you need to switch it to monthly subscription and specify storage space; (3) when the storage billing mode is monthly subscription and you need to adjust the storage space. For detailed directions, see [Adjusting Storage Space](#).

Example 3. Check the connection addresses of read-write instances, read-only instances, and database proxy in the cluster

1. Log in to the [TDSQL-C for MySQL console](#).
2. After login, if the cluster list page is in list view, you can toggle on the button in the top right corner to enable the console view mode, and switch from list view to tab view.



3. Find the target cluster in the cluster list and click it to enter the cluster management page.
4. On the **Cluster Management** page > **Cluster Details** tab, view the connection addresses of the corresponding instances. The read-write instance is on the left side, while the read-only instance is on the right side. If there are multiple read-only instances, you can click the read-only instance component to switch the query.
5. On the **Cluster Management** page > **Cluster Details** tab > **Database Proxy**, click **Details**.

M

Beijing Zone 3

Database Proxy [Details](#)

6. On the database proxy page redirected to, query the connection address set by the cluster database proxy under **Connection Address**.

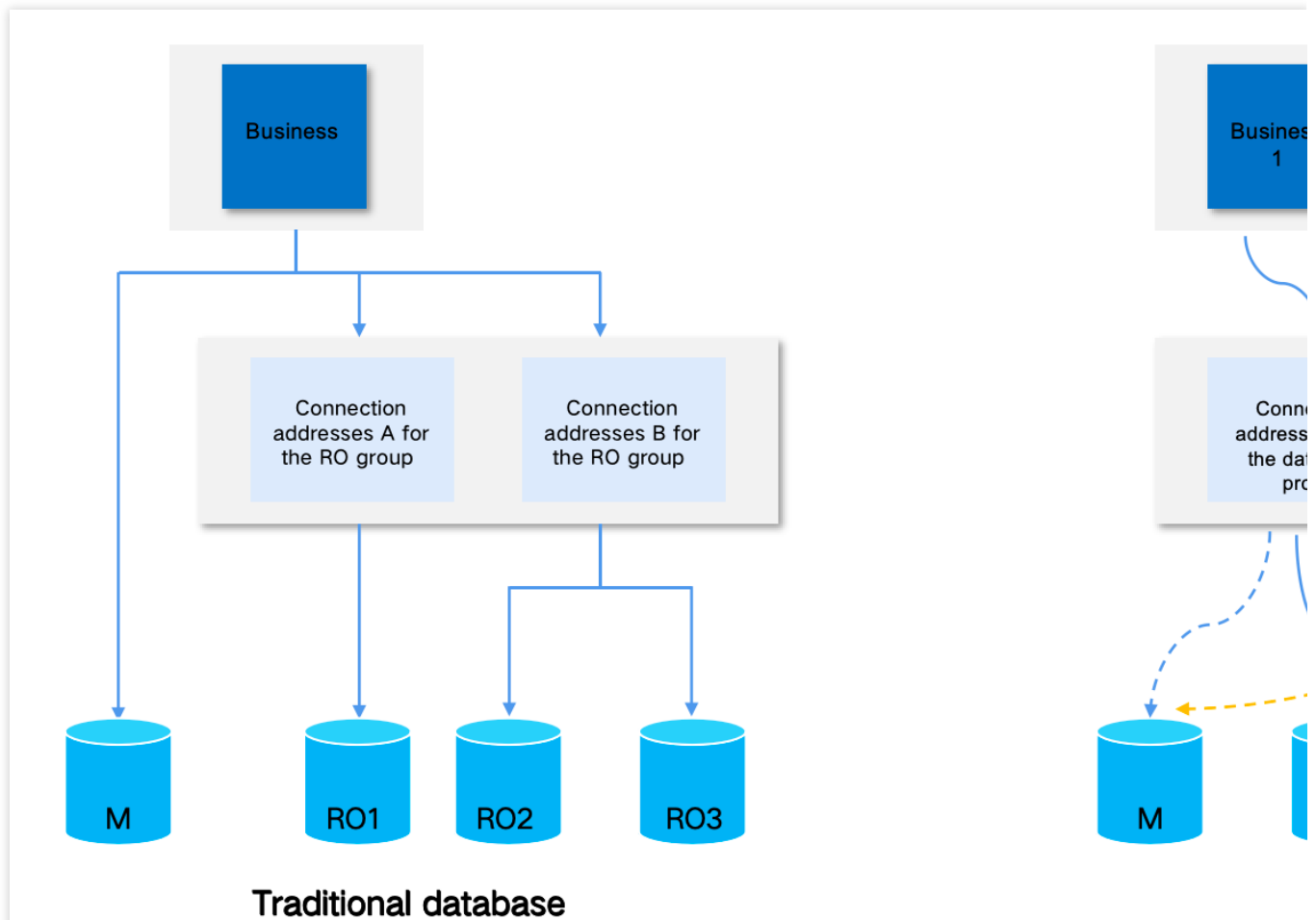
Private Network Access A...	Status	Read/Write Attribute	Connection Pool	Network	Remarks
IP: [redacted].8 Port:3306	Running	Read/Write Separation	Enabled(Session-Level Connection Pool)	Default-VPC Default-Subnet	--

Implementing Multiple RO Groups with Multiple Database Proxy Connection Addresses

Last updated : 2023-11-09 14:17:52

Background

An RO group is a set of read-only instances that can be configured with weights for traffic load balancing. You can create one or multiple RO groups for a database. By deploying RO groups as needed, you can send read requests to the corresponding read-only instances based on certain rules, significantly improving the read capacity of the database. The TDSQL-C for MySQL database proxy supports setting multiple proxy connection addresses, allowing different read-only instances to be mounted at different addresses. You can associate multiple proxy connection addresses with the same cluster as needed on your application, enabling the implementation of multiple RO groups scenario similar to traditional databases and providing additional advantages.



Multiple RO groups in traditional database

Example:

Source instance [connection address]: 10.0.1.22:3306.

RO group [connection address A]: 10.0.1.78:3306;

mounted RO instance: RO1.

RO group [connection address B]: 10.0.1.80:3306;

mounted RO instances: RO2, RO3.

Multiple connection addresses for database proxy

Example:

Database proxy [connection address A]: 172.20.0.10:3306;

mounted RO instance: RO1.

Database proxy [connection address B]: 172.20.0.12:3306;

mounted RO instances: RO2, RO3.

Note:

The above are examples of the IP address, RO group private network address, and database proxy connection address of the source instance, and no access connection service is provided.

After enabling the TDSQL-C MySQL database proxy, you can configure multiple database proxy connection addresses to mount different read-only instances, so that you can achieve multiple RO groups scenario similar to

traditional databases. (**Database proxy connection addresses A and B are equivalent to the RO group connection addresses A and B respectively in traditional database**). Additionally, compared to traditional databases with multiple RO groups, this setup offers the following advantages:

Support mounting multiple read-only instances, which outnumbers the traditional databases.

Support mounting with the read-write instances.

Support transaction split

Support connection pool

Support load rebalancing.

Support setting a consistency level

Support setting access mode

Mounting different instances through the database proxy connection address

Step 1. Enable the database proxy

For more information, see [Enabling Database Proxy](#).

Note:

With the database proxy enabled, you can create as many connection addresses as there are nodes. To achieve a multiple RO groups scenario, you must configure at least two nodes when enabling the database proxy.

Step 2. Mount an instance

Mounting an instance at a single database proxy connection address

After enabling the database proxy in TDSQL-C for MySQL, a proxy connection address will be provided by default, referred to as proxy connection address A. You can find this address by navigating to the **Database Proxy** tab >

Overview > Connection Address.

Cluster Details
Monitoring and Alarms
Account Management
Database Management
Database Proxy
Param

Overview
Access Policy
Performance Monitoring

Basic Info
[Disable Database Proxy](#)

Status/Task **Running**




Region/AZ North China region(Beijing)/Beijing Zone 3

Proxy Version 1.3.2 [Upgrade Kernel Minor Version](#)




Node Count 3 [Adjust Configurations](#)

Node Specification 2-core, 4000 MB MEM

Proxy Node

Node ID	Connections	Node Specification
cynosdbmysql-proxyNode- 	1	2-core, 4000 MB ME
cynosdbmysql-proxyNode- 	1	2-core, 4000 MB ME
cynosdbmysql-proxyNode- 	1	2-core, 4000 MB ME

Connection Address [Add Access Address](#) (1/3) ⓘ

Private Network Access A...	Status	Read/Write Attribute	Connection Pool	Network
IP:  .8 Port: 3306 	Running	Read/Write Separation	Enabled(Session-Level Connection Pool)	


Below is a description of the steps to mount the read-only instance RO1 at the proxy connection address A.

1. You can also find the target access address in **Database Proxy** tab > **Overview** > **Connection Address**. Then, click **Adjust Configurations** in the **Operation** column.
2. In the pop-up window, set **Read/Write Attribute** as **Read-Only**, complete the configuration for other policies, and click **Next**.
3. On the **Assign Read Weight** field, enable the read-only instance RO1, set its weight as needed, and click **OK**.

Mounting instances at multiple database proxy connection addresses

After mounting the read-only instance at a single database proxy connection address, you can also create multiple database proxy connection addresses to mount instances. Below is a description of how to add a new database proxy connection address B under the database proxy and then mount read-only instances RO2 and RO3.

1. On the **Database Proxy** tab, and click **Add Access Address** **next to** Connection Address.

Connection Address Address (1/3) ⓘ				
Private Network Access A...	Status	Read/Write Attribute	Connection Pool	Network
IP: .8 Port:3306	Running	Read/Write Separation	Enabled(Session-Level Connection Pool)	

2. In the **Create Connection** window, complete network configurations and click **Next**.

3. In the configuration policy window, set **Read/Write Attribute** as **Read-Only**, complete the configuration for other policies, and click **Next**.

4. On the **Assign Read Weight** field, enable the read-only instances RO2 and RO3, set their weights as needed, and click **OK**.

Step 3. Configure the application

After enabling the database proxy and mounting instances at the proxy connection addresses, you can configure the database proxy addresses A and B separately for different business purposes. By using these proxy addresses, requests can be routed through the proxy cluster to different instances within the access cluster. This achieves the functionality provided by traditional databases with multiple RO groups and offers additional advantages.

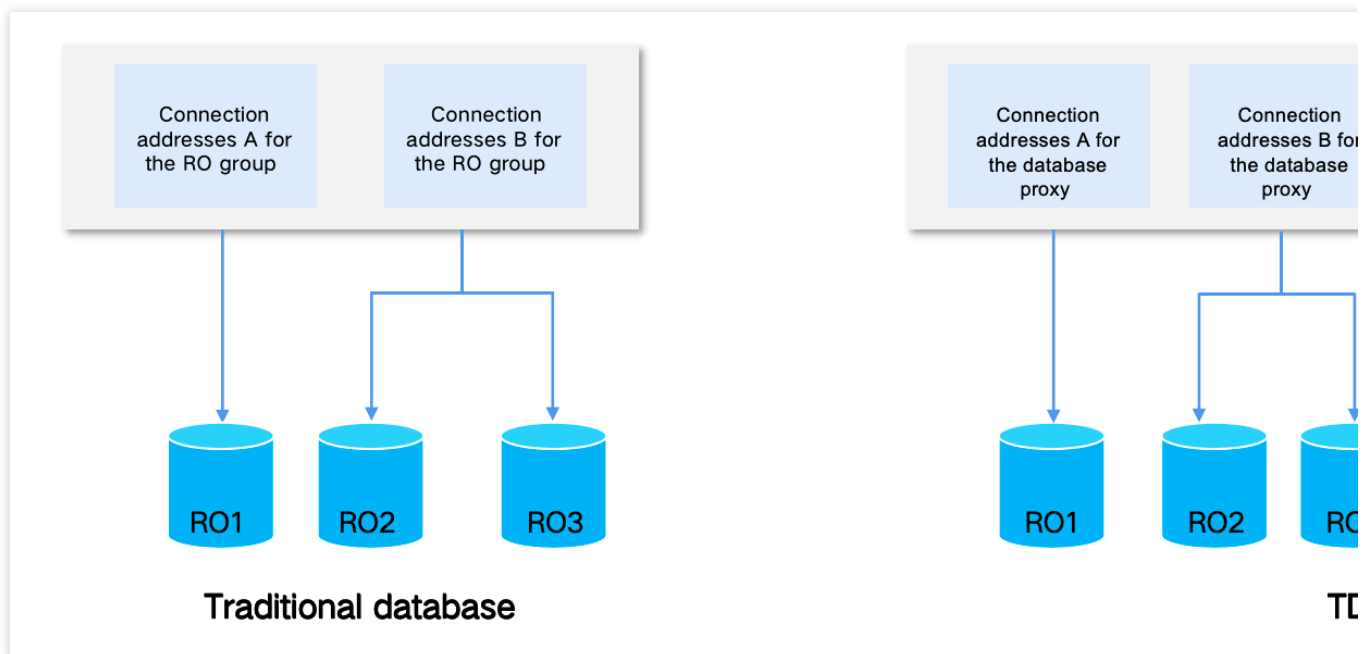
Strengths of Database Proxy

Last updated : 2023-11-09 10:38:10

This document describes the database proxy capability of TDSQL-C for MySQL. Compared to traditional databases with multiple RO groups, the key advantage of TDSQL-C MySQL is that it reduces the load on the source instance.

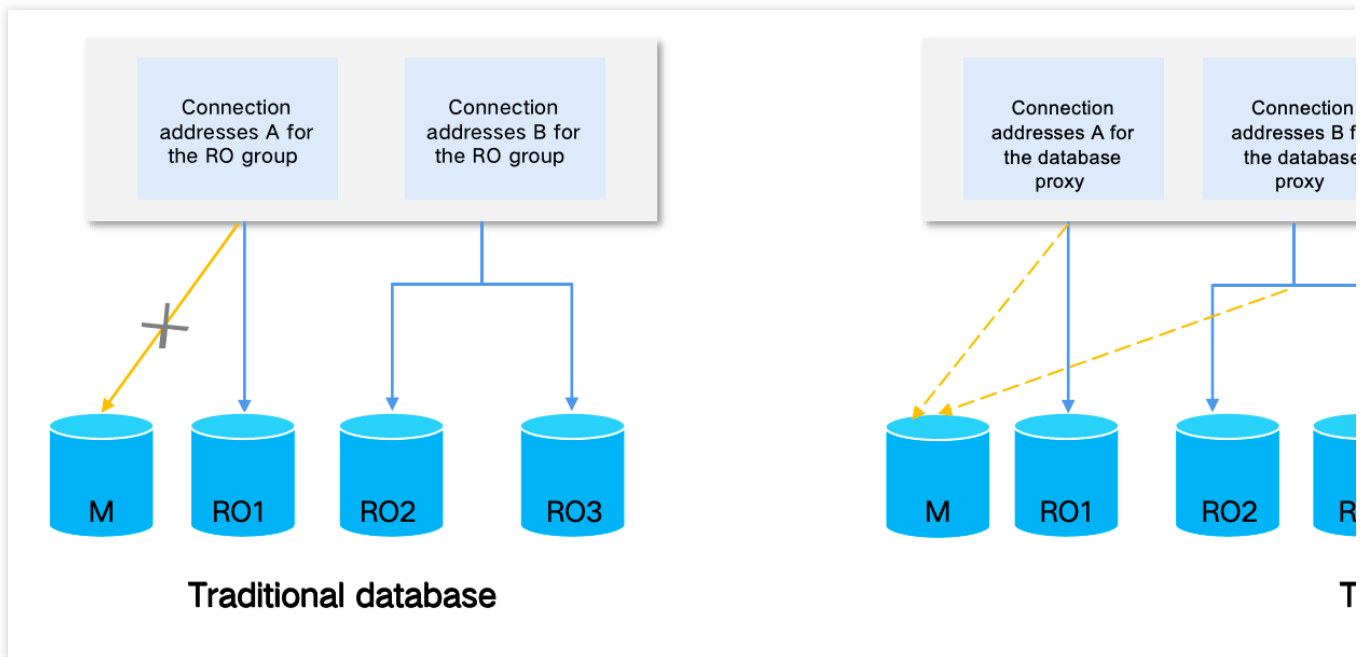
Support multiple independent database proxy connection addresses

In traditional databases, a maximum of two RO groups can be created for a database, which may not be enough to meet the demands of various business loads. However, in TDSQL-C or MySQL, the database proxy feature allows for the creation of as many proxy connection addresses as there are nodes. The current database version supports up to four nodes.



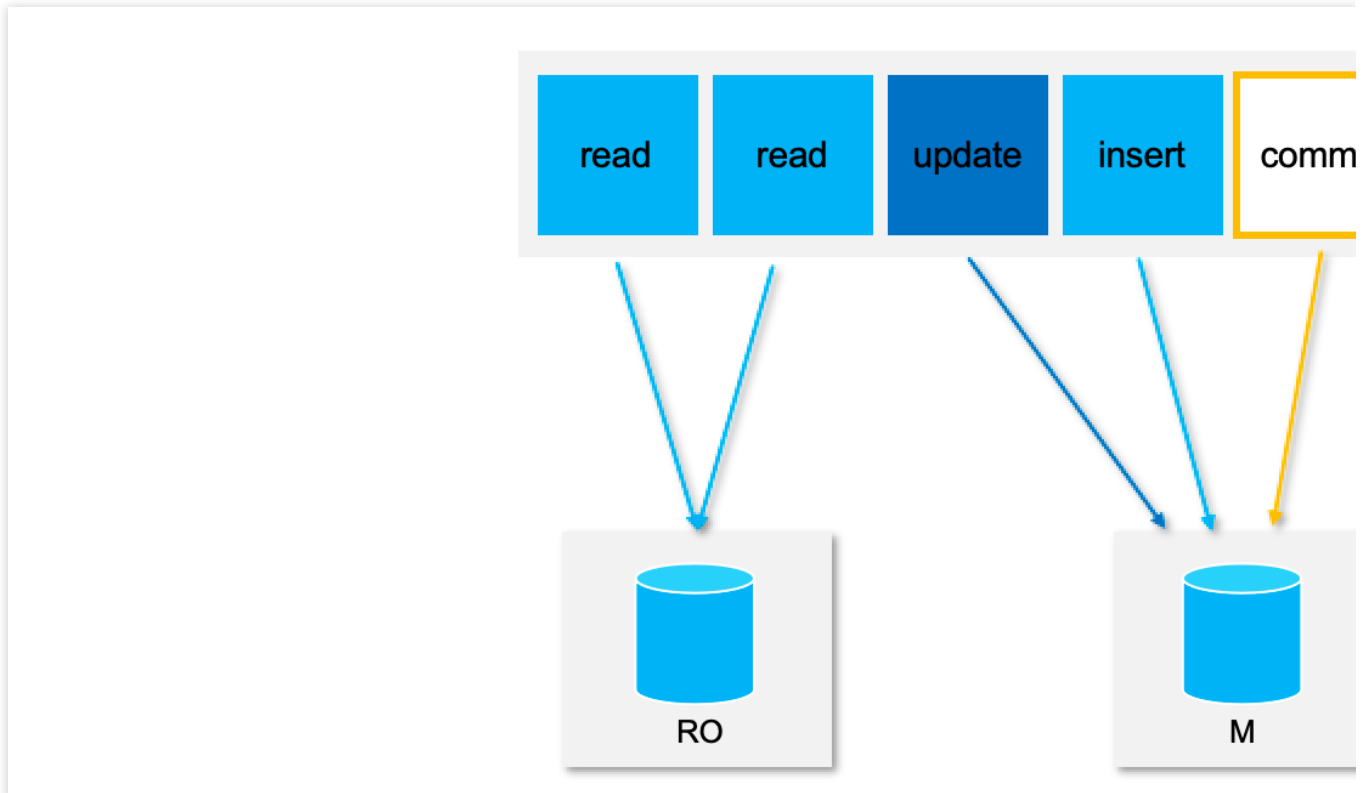
Support mounting with the read-write instance

In traditional databases, read-only instances can only be mounted within the RO group and cannot be mounted with the read-write instance. However, in TDSQL-C for MySQL, the read-only instances can be mounted with the read-write instance at each database proxy address. This allows for access balancing to both the read-write instance and the read-only instances through the proxy address.



Support transaction split

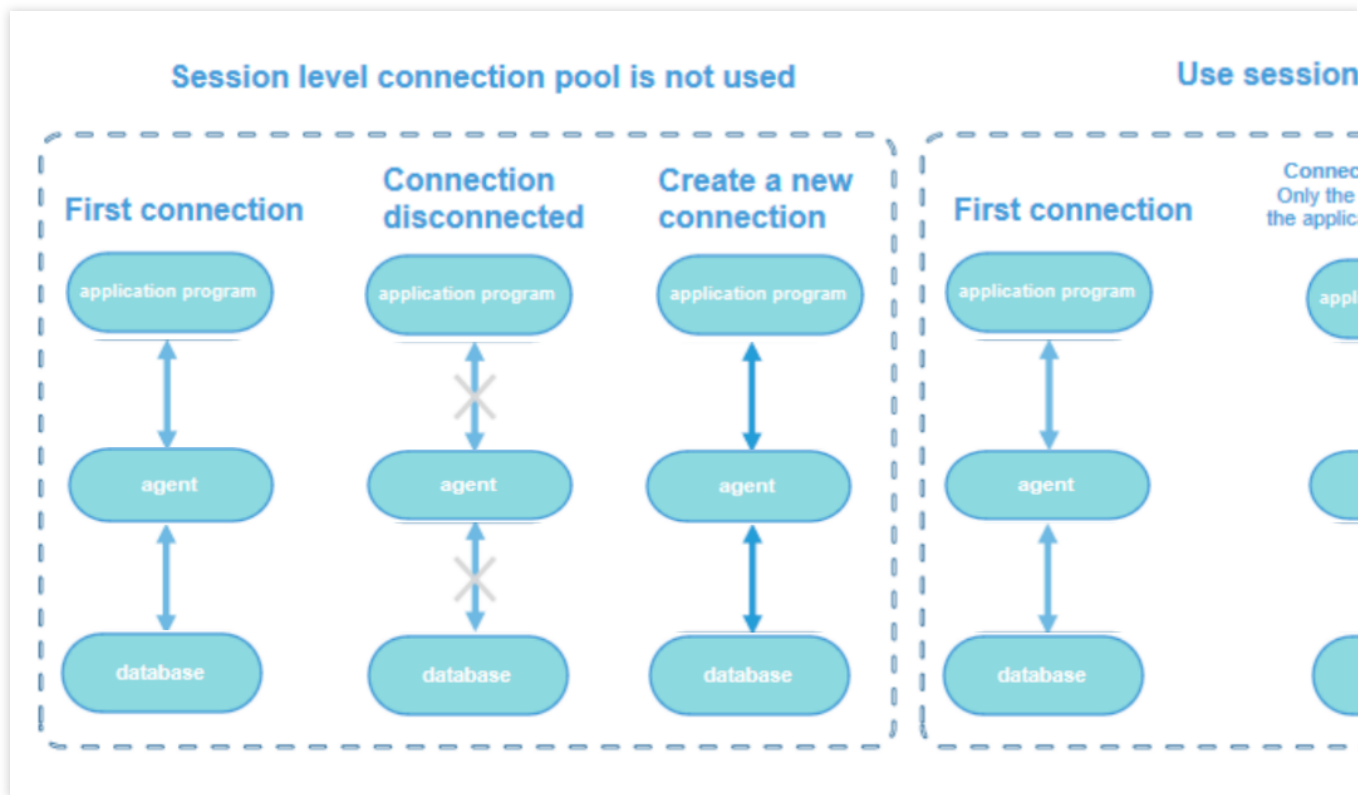
The TDSQL for MySQL database proxy provides the transaction split feature. This feature separates read and write operations in one transaction to different instances for execution and forwards read requests to read-only instances, thereby lowering the load of the source instance.



Support session-level connection pool

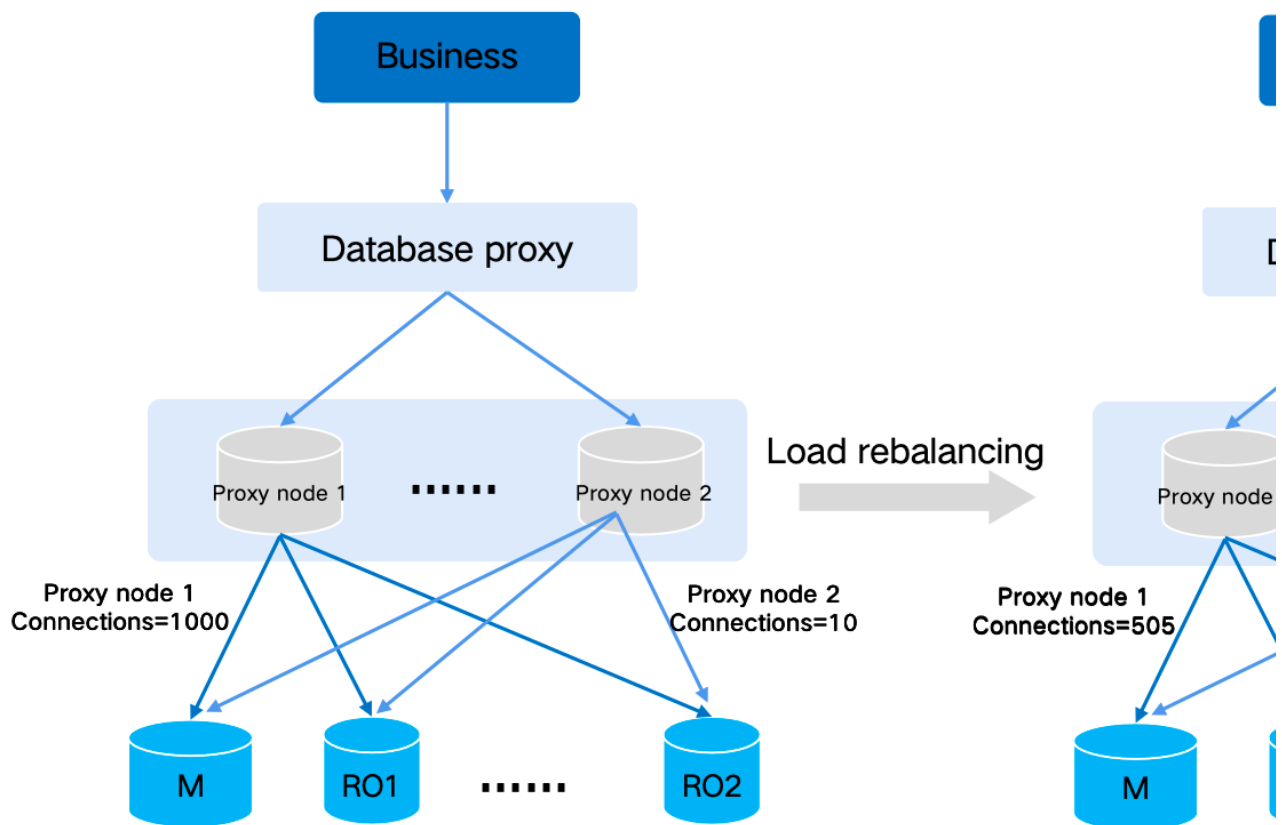
The TDSQL-C for MySQL database proxy supports the session-level connection pool feature. It can effectively solve the problem of excessively high database instance loads caused by frequent establishments of new non-persistent

connections. If a client connection is closed, the system will determine whether the current connection is idle, and if so, the system will put it into the proxy connection pool and retain it for a short period of time, which is five seconds by default and can be customized. For more information, see [Setting Session-Level Connection Pool](#).



Support load rebalancing

After enabling the database proxy, you can view **Connections** in the proxy node list or view the performance monitoring data of each proxy node to check whether the numbers of connections on the nodes are unbalanced. If there are a large number of persistent connections in the business, adding more database proxy nodes may result in uneven node loads. If there is an imbalance in the connection count among the proxy nodes, you can redistribute the connections through load balancing to achieve a more balanced distribution.



Support setting consistency levels

When there are data updates on the read-write instance, the updates will be applied to the read-only instances. The delay in data sync depends on the write workload. To ensure the consistency requirements of accessing database data, TDSQL-C for MySQL provides three different consistency levels as follows

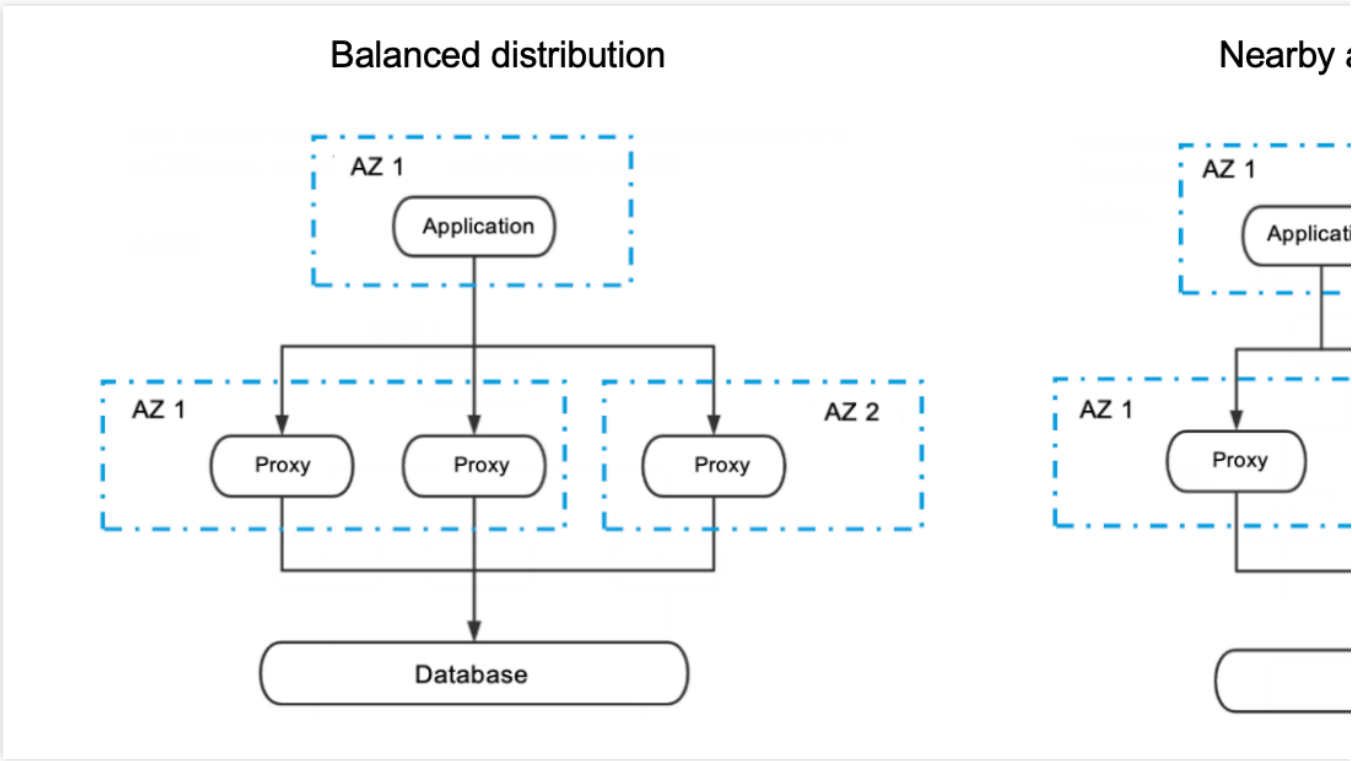
Eventual consistency: Data can achieve eventual consistency, ensuring that read-only instances can read the updated data. All updated data can be obtained eventually, but immediate access is not guaranteed. Due to the delay in source-replica replication, the results obtained from different nodes may differ when querying the updated data.

Session consistency: This guarantees monotonic reads in the same session, and the data updated before the execution of a read request can be queried.

Global consistency: In the same session, the data updated before the execution of a read request can be queried. Besides, the query results are consistent for requests sent through different connections.

Support setting access mode

You can set an access mode to control the connection link between the application/client and the database proxy. There are two access modes for your choice: load balancing and nearby access. The load balancing mode achieves balanced distribution of traffic, eliminating the issue of a single node being overloaded. In nearby access mode, the application connects to the database proxy node that is in the same AZ or closest to it. If there are multiple proxy nodes in the same AZ, the application will still choose the one that is closest to it. This mode has the benefits of low latency and fast speed.



Selecting Billing Mode for Storage Space

Last updated : 2023-05-31 16:10:40

Storage space refers to the space used by data files, index files, log files (redo logs, undo logs, slow logs, and error logs), and temporary files. Fees are charged for the used storage space.

This document describes how to select a billing mode for the storage space.

Pay-as-You-Go

Billing rules

The storage space of TDSQL-C for MySQL is billed per GB per hour. You don't need to select the storage space capacity when purchasing an instance, as the storage space will be automatically expanded as the data volume grows. You only pay for the storage space you use. You can view the database storage usage details and the storage space cap under the current compute node specification in the configuration information on the cluster details page in the [console](#).

Cluster Details

Instance List

Account Management

Database Management

Parameter Settings

Security Group

Backup Management

Basic Info

Cluster Name:

Status: Running

Project: DEFAULT PROJECT [Switch to Another Project](#)

Region/AZ:

Cluster ID:

Network: [Change Network](#)

Tag:

Configuration Info

Compatible Database: MySQL 5.7

Database Storage (Used/Total): 16MB / 1T(Max storage supported by the current specification: 3T)

Database Version: 2.0.15 [Upgrade](#)

Binlog Settings:

Enable

Pricing

Guangzhou, Shanghai, Beijing, and Nanjing: 0.00072 USD/GB/hour.

Hong Kong (China), Taipei (China), Singapore, Silicon Valley, Frankfurt, Tokyo, and Virginia: 0.000792 USD/GB/hour.

Use cases

It is suitable for instantaneously fluctuating businesses. In this mode, instances can be released immediately after the use to save costs.

Monthly Subscription

Billing rules

The monthly subscribed storage space can be selected only when the billing mode of TDSQL-C for MySQL compute nodes is monthly subscription. Compared with the pay-as-you-go billing, the (prepaid) monthly subscription billing mode offers discounts when your required storage space is large.

Pricing

Guangzhou, Shanghai, Beijing, and Nanjing: 0.20541177 USD/GB/month for below 3000 GB; 0.18829412 USD/GB/month for 3000 GB or above.

Hong Kong (China), Taipei (China), Singapore, Silicon Valley, Frankfurt, Tokyo, and Virginia: 0.22447059 USD/GB/month for below 3000 GB; 0.20576471 USD/GB/month for 3000 GB or above.

Use cases

It is more cost-effective in the long term for businesses with stable needs than pay-as-you-go billing. It supports tiered pricing with a higher cost performance.

Storage Space Price Comparison

Note:

The actual daily usage cannot be predicted in pay-as-you-go mode, which makes comparison difficult. Therefore, the following pay-as-you-go storage space usage is based on the amount of stored data imported at one time within a month.

Storage Space (GB)	Chinese Mainland		Hong Kong (China), Taipei (China), and other countries and regions	
	Pay-as-You-Go (USD/Month)	Monthly Subscription (USD/Month) Equivalent to 36% to 40% of Pay-as-You-Go Prices	Pay-as-You-Go (USD/Month)	Monthly Subscription (USD/Month) Equivalent to 36% to 39% of Pay-as-You-Go Prices
50	25.92	10.2705885 (60% off)	28.512	11.2235295 (61% off)
100	51.84	20.541177 (60% off)	57.024	22.447059 (61% off)
200	103.68	41.082354 (60% off)	114.048	44.894118 (61% off)
300	155.52	61.623531 (60% off)	171.072	67.341177 (61% off)
500	259.2	102.705885 (60% off)	285.12	112.235295 (61% off)
1000	518.4	205.41177 (60% off)	570.24	224.47059 (61% off)

2000	1036.8	410.82354 (60% off)	1140.48	448.94118 (60% off)
3000	1555.2	564.88236 (64% off)	1710.72	617.29413 (64% off)
5000	2592	941.4706 (64% off)	2851.2	1028.82355 (64% off)
10000	5184	1882.9412 (64% off)	5702.4	2057.6471 (64% off)
400000	207360	75317.648 (64% off)	228096	82305.884 (64% off)

Creating Remote Disaster Recovery by DTS

Last updated : 2023-11-22 14:47:43

TDSQL-C for MySQL allows you to use Data Transmission Service (DTS) to implement remote disaster recovery to ensure database availability and stability.

Overview

If your TDSQL-C for MySQL cluster is deployed in a single region, your service is more likely to be interrupted by force majeure like power outages and network disconnection.

To prevent this problem, you can build a disaster recovery center in another region to improve service availability.

DTS allows you to continually sync data and replicas between your business center and disaster recovery center. In case of a failure in your business region, you can switch to the disaster recovery region to process user requests.

With the remote disaster recovery architecture, if a disaster occurs in an IDC, the traffic in this IDC can be routed to other IDCs to implement quick cross-region failover and guarantee business continuity.

Remote Disaster Recovery Directions

Step 1. Purchase a cluster

Go to the [TDSQL-C for MySQL purchase page](#) and purchase two TDSQL-C for MySQL clusters in two different regions, with one deployed in your business center and the other in the disaster recovery center.

Note:

For data sync prerequisites and environment requirements, see [Sync from TDSQL-C for MySQL to TDSQL-C for MySQL](#).

Step 2. Use DTS to implement remote disaster recovery

1. Go to the [data sync purchase page](#), select appropriate configuration items, and click **Buy Now**.

Parameter	Description
Billing Mode	Pay-as-you-go billing and monthly subscription are supported.
Source Instance Type	Select "TDSQL-C for MySQL", which cannot be changed after purchase.
Source Instance Region	Select the source instance region, which cannot be changed after purchase.

Target Instance Type	Select "TDSQL-C for MySQL", which cannot be changed after purchase.
Target Instance Region	Select the target instance region, which cannot be changed after purchase.
Specification	Select a specification based on your business needs. The higher the specification, the higher the performance. For more information, see Billing Overview .

2. After successful purchase, return to the [data sync task list](#), and you can see the newly created data sync task. You need to configure it before using it.

3. In the data sync task list, click **Configure** in the **Operation** column to enter the sync task configuration page.

4. On the sync task configuration page, configure the source and target instances and their accounts and passwords, test the connectivity, and click **Next**.

As there are many overlapped scenarios of source database deployment modes and access types, the sync steps for different scenarios are similar. Below is a configuration example for typical scenarios, to which you can refer for other scenarios.

Sync between TDSQL-C for MySQL instances is used as an example.

Category	Parameter	Description
Task Settings	Task Name	DTS will automatically generate a task name, which is customizable.
	Running Mode	Immediate execution and scheduled execution are supported.
Source Instance Settings	Source Instance Type	Select the source instance type selected during purchase, which cannot be changed once configured.
	Source Instance Region	Select the source instance region selected during purchase, which cannot be changed once configured.
	Service Provider	For a self-built database (such as a CVM-based one) or TencentDB database, select "Others". For a third-party cloud database, select the corresponding service provider. In this scenario, select "Others".
	Access Type	Select a type based on your scenario. In this scenario, select "Database". For the preparations for different access types, see Overview . Public Network: The source database can be accessed through a public IP. Self-Build on CVM: The source database is deployed on a CVM instance.

		<p>Direct Connect: The source database can be interconnected with VPCs through Direct Connect.</p> <p>VPN Access: The source database can be interconnected with VPCs through VPN Connections.</p> <p>Database: The source database is a TencentDB database.</p> <p>CCN: The source database can be interconnected with VPCs through CCN.</p> <p>VPC: The source and target databases are both deployed in Tencent Cloud VPCs. To use the VPC access type, submit a ticket for application.</p>
	Instance ID	Source instance ID. You can view the source instance information in the cluster list .
	Account	Account of the source instance, which must have the required permissions.
	Password	Password of the source instance account.
Target Instance Settings	Target Instance Type	The target instance type selected during purchase, which cannot be changed.
	Target Instance Region	The target instance region selected during purchase, which cannot be changed.
	Access Type	Select a type based on your scenario. In this scenario, select "Database".
	Instance ID	Target instance ID.
	Account	Account of the target instance, which must have the required permissions.
	Password	Password of the target instance account.

5. On the **Set sync options and objects** page, set the data initialization, data sync, and sync object options and click **Save and Go Next**.

Note:

If you only select **Full data initialization** for **Initialization Type**, the system will assume by default that you have created the table structures in the target database and will neither sync table structures nor check whether the source and target databases have tables with the same name. Therefore, if you select **Precheck and report error** for **If Target Already Exists**, the precheck and error reporting feature won't take effect.

If you want to rename a table (for example, rename table A "table B") during the sync, you must select the entire database (or entire instance) where table A resides rather than only table A as the **sync object**; otherwise, the system

will report an error.

Category	Parameter	Description
Data Initialization Option	Initialization Type	<p>Structure initialization: Table structures in the source database will be initialized into the target database before the sync task runs.</p> <p>Full data initialization: Data in the source database will be initialized into the target database before the sync task runs. If you only select Full data initialization, you need to create the table structure in the target database in advance. Both options are selected by default, and you can deselect them as needed.</p>
	If Target Already Exists	<p>Precheck and report error: If a table with the same name exists in both the source and target databases, an error will be reported, and the task will stop.</p> <p>Ignore and execute: Full and incremental data will be directly added to tables in the target instance.</p>
Data Sync Option	Conflict Resolution Method	<p>Report: If a primary key conflict is found during data sync, an error will be reported, and the data sync task will be paused.</p> <p>Ignore: If a primary key conflict is found during data sync, the primary key record in the target database will be retained.</p> <p>Overwrite: If a primary key conflict is found during data sync, the primary key record in the source database will overwrite that in the target database.</p>
	SQL Type	Supported operations include

		INSERT, UPDATE, DELETE, and DDL. If you select "Custom DDL", you can select different DDL statement sync policies as needed. For more information, see Setting SQL Filter Policy .
Sync Object Option	Database and Table Objects of Source Instance	<p>Select the objects to be synced. You can select basic databases, tables, views, procedures, and functions.</p> <p>The sync of advanced objects is a one-time operation: only advanced objects already in the source database before the task start can be synced, while those added to the source database after the task start will not be synced to the target database. For more information, see Syncing Advanced Object.</p>
	Selected Object	<p>Database/Table mapping (renaming) is supported. Hover over a database or table name, click the displayed "Edit" icon, and enter a new name in the pop-up window.</p> <p>When advanced objects are selected for sync, we do not recommend that you rename databases/tables. If you do so, sync of the advanced objects may fail.</p>
	Sync Online DDL Temp Table	<p>If you use the "gh-ost" or "pt-osc" tool to perform online DDL operations on source database tables, DTS supports migrating the generated temp tables to the target database.</p> <p>If you select `gh-ost`, DTS will migrate the temp tables named `_Table name_ghc`, `_Table name_gho`, and `_Table name_del` to the target database.</p>

		If you select "pt-osc", DTS will migrate the temp tables named `_Table name_new` and `_Table name_old` to the target database. For more information, see Syncing Online DDL Temp Table .
--	--	--

6. On the task verification page, complete the verification. After all check items are passed, click **Start Task**.

If the verification fails, troubleshoot as instructed in [Check Item Overview](#) and initiate the verification again.

Failed: It indicates that a check item failed and the task is interrupted. You need to fix the problem and run the verification task again.

Alarm: It indicates that a check item doesn't completely meet the requirements. The task can be continued, but the business will be affected. You need to assess whether to ignore the alarm or fix the problem and continue the task based on the alarm message.

7. Return to the data sync task list, and you can see that the task has entered the **Running** status.

Note:

You can click **More > Stop** in the **Operation** column to stop a sync task. Before doing so, ensure that data sync has been completed.

8. (Optional) You can click a task name to enter the task details page and view the task initialization status and monitoring data.

Creating VPC for Cluster

Last updated : 2023-09-12 14:49:53

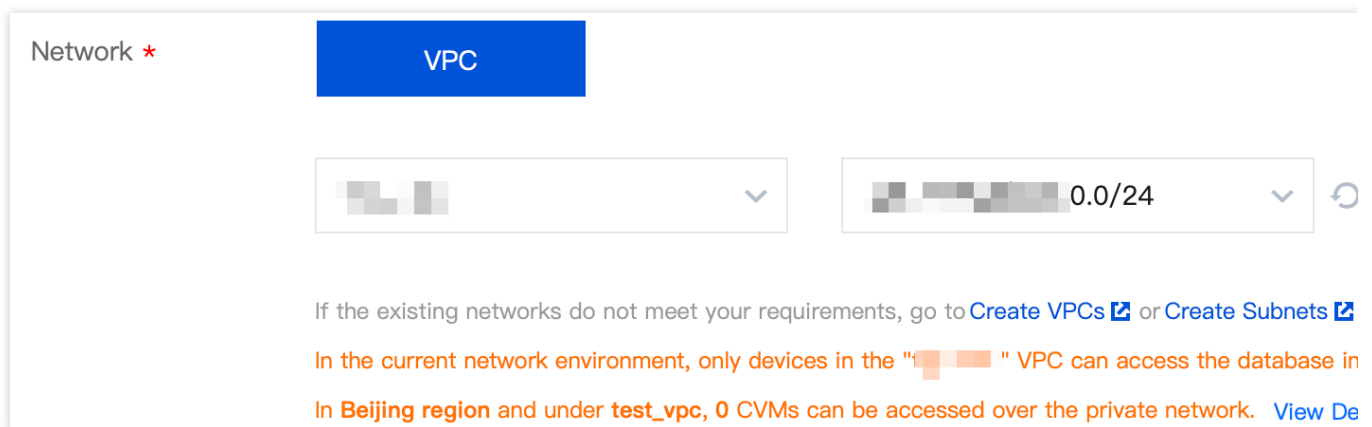
Tencent Cloud provides [Virtual Private Cloud \(VPC\)](#), a platform for hosting TencentDB instances. You can launch Tencent Cloud resources in a VPC, such as TencentDB instances.

A common scheme is to share data between a TencentDB instance and a web server running in the same VPC. This document uses this scheme to create a VPC and add a TencentDB instance to it.

This document describes how to add CVM and TDSQL-C for MySQL instances in the same VPC to interconnect each other over the private network.

Background

Your Tencent Cloud account has not created a VPC. When purchasing a TDSQL-C MySQL instance, you cannot select other VPCs and subnets in the network option. In this case, you can follow the instructions below to create a new VPC.



To access a TDSQL-C MySQL instance through a CVM instance, the two instances must be in the same VPC. If their VPCs are different, you can use add the CVM instance and the database under the same VPC to ensure normal access to the cluster.

Step 1. Create a VPC

A VPC has at least one subnet, and Tencent Cloud service resources can only be added in a subnet.

1. Log in to the [VPC console](#).
2. Select the region of the VPC at the top of the list and click **Create**.

3. Enter the VPC information and initial subnet information and click **OK**. The CIDRs of the VPC and subnet cannot be modified after creation.

The VPC CIDR can be any of the following IP ranges. If you want two VPCs to communicate with each other over the private network, their CIDRs should not overlap.

10.0.0.0 - 10.255.255.255 (the mask range must be 12-28)

172.16.0.0 - 172.31.255.255 (mask range between 12 and 28)

192.168.0.0 - 192.168.255.255 (mask range between 16 and 28)

The subnet CIDR block must be within or the same as the VPC CIDR block.

For example, if the IP range of a VPC is 192.168.0.0/16, then that of its subnets can be 192.168.0.0/16, 192.168.0.0/17, etc.

Create VPC

VPC information

Region North China region(Beijing)

Name

Up to 60 characters ([a-z], [A-Z], [0-9], [-_]) and Chinese

IPv4 CIDR
Block

 . . . /

The IP range cannot be changed once created. It's recommended to plan a proper [network structure](#).

Tags

Tag Value

[+ Add](#)

Subnet information

Subnet name

Up to 60 characters ([a-z], [A-Z], [0-9], [-_.] and Chinese)

IPv4 CIDR Block

10 . 0 .

0

 . 0 /

24 ▼

Remaining IPs: 253

Availability zone ⓘ

Please select ▼

Associated route table

Default ⓘ

Tags

Tag Key ▼

Tag Value

+ Add

OK


Close


Step 2. Create a subnet

You can create one or more subnets at a time.

1. Log in to the [VPC console](#).
2. Click **Subnet** on the left sidebar to enter the management page.
3. Select the region and VPC in which the subnet is to be created and click **Create**.
4. Enter the subnet's name, CIDR, AZ, and associated route table.

Create a subnet

Network  0/16) ▾ 1 existing subnets

Subnet name	VPC IP range	CIDR ⓘ	Availability zone ⓘ
<input type="text" value="Enter the subnet name."/>	0/60  0.0/16 ▾	10 . 0 . 0 . 0 / 24 ▾	Please select ▾
+ New line			

[Advanced options ▸](#)

[Create](#) [Cancel](#)

5. (Optional) Click **+New line** to create multiple subnets at a time.

6. Click **Create**.

Step 3. Create a route table and associate it with a subnet

You can create a custom route table, edit its routing policy, and associate it with a specified subnet. The route table associated with a subnet is used to specify the outbound routes for the subnet.

1. Log in to the [VPC console](#) and select **Route Tables** on the left sidebar.
2. Select the region and VPC at the top of the list and click **Create**.
3. In the pop-up dialog box, enter the name, network, routing rules and click **Create**. Return to the route table list to view the newly created route table.

Create route table

Name
60 more characters allowed

Network

Tags ×
[+ Add](#)

Routing rules

ℹ Routing policies control the traffic flow in the subnet. For details, please see [Configuring Routing Policies](#).

Destination	Next hop type	Next hop	Remark
Local	LOCAL	Local	Delivered by default
<input type="text" value="such as 10.0.0.0/16"/>	<input type="text" value="Public IP of CVM"/>	Public IP of CVM ℹ	<input type="text"/>

[+ New line](#)

[Create](#) [Close](#)

4. Click **Subnet** on the left sidebar, select the subnet to be associated with the route table, and click **More > Change route table** in the **Operation** column to associate it.

Virtual Private Cloud

- Network Topology Map
- Network Performance Dashboard
- Virtual Private Cloud
- Subnet**


Subnet
Beijing 2
All VPCs

[Create](#)

ID/Name	Network	CIDR	Availability zone	Associated route table	CVM	Available IPs
subnet-test_s	vpc-test	10.0.0.0/24	Beijing Zone 3	rtb-C	0	248
subnet-Default	vpc-Default	10.0.0.0/20	Beijing Zone 3	rtb-f	0	4082

Step 4. Add a CVM instance

1. Log in to the [VPC console](#).
2. Click **Subnet** on the left sidebar to enter the management page.
3. Select the target subnet, click the icon in the **CVM** column to add a CVM.

ID/Name	Network	CIDR	Availabili... ▼	Associated r...	CVM	Available IPs	Defau
subnet-5f611111 test_	vpc-5f611111 test	0.0/24	Beijing Zone 3	rtb-5f611111 -		248	No

4. Complete the purchase of the CVM instance as instructed on the page. For more information, see [Purchasing Channels](#).


Step 5. Add a TencentDB instance


New database

1. Log in to the [TDSQL-C for MySQL console](#) and click **Create** in the instance list to enter the purchase page.
2. In the **Network** section on the purchase page, select the previously created VPC and the corresponding subnet, and add the new TencentDB instance to the VPC.

Network *

VPC


▼

 0.0/24
▼

↻


If the existing networks do not meet your requirements, go to [Create VPCs](#) or [Create Subnets](#).





In the current network environment, only devices in the "test_vpc" VPC can access the database in Beijing region and under test_vpc, 0 CVMs can be accessed over the private network. [View De](#)


Existing database


1. On the [cluster list](#), click the **cluster ID** to go to the **Cluster Management** page.
2. Select **Cluster Management** > **Cluster Details**. In the **Network** section on the details page of the read-write instance or the read-only instance, switch to the corresponding VPC.

Read-Write Instance




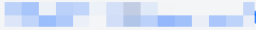



Instance ID cynosdbmysql- [Details](#)

Instance Name cynosdbmysql-i- [Edit](#)

Configuration 2-core/4 GB MEM [Adjust Configurations](#)




Status ● Running 

Network  [Change Network](#)

Expiration Time ● 2023-09-04 10:26:31  [Renewal Management](#)

Read/Write Address


Private





Host .1 Port 3306  

Public

-- [Enable](#)

Read-Only Instance





No read-only in

[+Add Rea](#)

Data Rollback

Database/Table-Level Rollback

Overview

Last updated : 2023-10-08 14:42:25

When a table or data in the cluster is accidentally deleted or modified, if you need restore it to its original status, you can restore the data to the original cluster.

TDSQL-C for MySQL supports database/table-level rollback by time point and backup set.

Rollback by time point: The selection range of time point is determined by the log backup retention period you set.

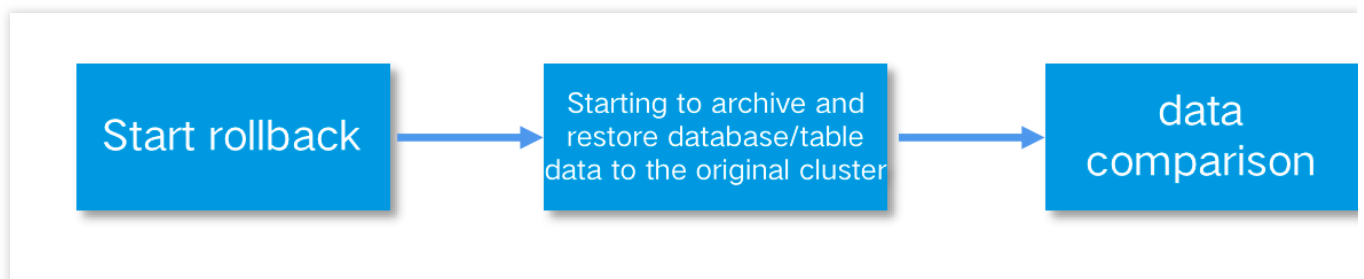
Rollback by backup set: The selection range of the backup file is determined by the data backup retention period you set.

After a cluster is created, TDSQL-C for MySQL generates backup files every 6-48 hours on a 24/24 uninterrupted basis, which has no impacts on the instance performance. You can also set the backup retention time in the console as needed. Based on this, when you perform database/table-level rollback on the cluster, the available time points and backup sets will be directly displayed for your selection in the console.

Process

To roll back by time point or by backup set, you need to roll back the backup data to the original cluster first, and then compare the data.

The process is as shown below:



Instructions

[Database/Table Rollback by Time Point](#)

[Database/Table Rollback by Backup Set](#)

Database/Table Rollback by Time Point

Last updated : 2023-09-12 14:54:33

Through a complete data backup and a subsequent period of binlog backup, you can roll back a specific database/table to any time point. After confirming the time point when the misoperation occurred, you can use this method to restore the misoperated database or table to the status before the misoperation occurred.

Prerequisites

There are data backup files and log backup files in the backup list. For automatic backup settings, see [Automatic Backup](#).

The cluster is in the running status.

Note

For database/table rollback, you need to specify the database/table to be rolled back. If you cannot determine all the involved databases/tables, we recommend that you clone the original cluster and migrate back to it after determining the databases/tables.

If the database/table to be rolled back does not exist at the specified time point, database/table rollback will fail.

If the database/table to be rolled back does not exist or has been dropped, you need to log in to the database and create a database/table first before performing rollback in the console.

If there are primary or foreign key constraints in the specified database/table to be rolled back, please ensure that the associated databases/tables exist during the rollback process; otherwise, database/table rollback will fail.

Up to 500 databases or tables can be rolled back at a time.

Step 1. Obtain the time point of misoperation

If you have enabled the database audit service for TDSQL-C for MySQL cluster, you can determine the time point of SQL execution misoperation by analyzing the audit log.

Step 2. Roll back databases/tables

1. Log in to the [TDSQL-C for MySQL console](#).
2. Select a region at the top and proceed according to the actually used view mode.

Tab view

List View

Click **Target Cluster** in the cluster list on the left to enter the cluster management page.

Find the target cluster in the cluster list and click the **Cluster ID** or **Manage** in the **Operation** column to enter the cluster management page.

3. On the cluster management page, select the **Backup Management** tab and click **Roll Back**.

4. On the database/table rollback page, set the rollback mode to **By time point**. Then, select the database/table to be rolled back, rename it, and click **Roll Back**.

Note:

Fast mode: Import full backup of the cluster, and then roll back the selected databases and tables. This rollback mode is slower than the other modes but has no limit.

Faster mode: Full backup + database-level binlog. For cross-database operation, if associated database is not selected at the same time, the database callback will fail.

Ultrafast mode: Full backup + table-level binlog. For cross-table operation, if associated table is not selected at the same time, the table rollback will fail.

5. In the pop-up window, confirm that everything is correct and click **OK** to initiate the rollback task.

6. After the task is submitted, you can click **View Rollback Task** or go to the task list to view the rollback progress and task details.

7. After the rollback is completed, you can see the new restored database/table in the original cluster and perform further operations.

Step 3. Compare the data

After the databases/tables are rolled back, you can log in to the cluster through DMC to compare and verify the data.

1. Log in to the [TDSQL-C for MySQL console](#).

2. In the cluster list, find the rollback cluster and click **Log In**.

3. On the login page, enter the cluster account and password, and click **Log In**.

4. After entering the DMC management page, select the rollback database from the drop-down menu on the left and find the misoperation data. Then, verify that the data has been rolled back to its original status and other data is also consistent with how it was before the misoperation.

Database/Table Rollback by Backup Set

Last updated : 2023-09-12 14:55:44

If you can't specify a time point for data rollback, you can use a backup set to restore the database to a certain status at a faster rate.

Prerequisites

There are data backup files in the backup list. For automatic backup settings, see [Automatic Backup](#).

The cluster is in the running status.

Note

For database/table rollback, you need to specify the database/table to be rolled back. If you cannot determine all the involved databases/tables, we recommend that you clone the original cluster and migrate back to it after determining the databases/tables.

If the database/table to be rolled back does not exist at the specified time point, database/table rollback will fail.

If the database/table to be rolled back does not exist or has been dropped, you need to log in to the database and create a database/table first before performing rollback in the console.

If there are primary or foreign key constraints in the specified database/table to be rolled back, please ensure that the associated databases/tables exist during the rollback process; otherwise, database/table rollback will fail.

Up to 500 databases or tables can be rolled back at a time.

Step 1. Roll back databases/tables

1. Log in to the [TDSQL-C for MySQL console](#).
2. Select a region at the top and proceed according to the actually used view mode.

Tab view

List view

Click **Target Cluster** in the cluster list on the left to enter the cluster management page.

TDSQL-C Cluster List Beijing 3 Other regions 12

<< Hide cluster lists (3) Create Quick Check Renew More Separate keywords

cynosdbmysql-

cynosdbmysql-
Beijing Zone 3 | MySQL 5.7

cynosdbmysql-

cynosdbmysql-
Beijing Zone 3 | MySQL 5.7

cynosdbmysql- Renew

cynosdbmysql-
Beijing Zone 3 | MySQL 8.0

cynosdbmysql- Running

ID cynosdbmysql-

Database Version 2.1.9 Upgrade

Project Default Project Adjust

Renewal Settings Set

Transfer Linkage High IO

Tag

Billing Mode Compute: Monthly Subscription

Cluster Expiration Time 2023-09-04 10:26:31

Deployment Mode Single-AZ

Cluster Details Monitoring and Alarms Account Management Database Management Database Proxy Pa

Find the target cluster in the cluster list and click the **Cluster ID** or **Manage** in the **Operation** column to enter the cluster management page.

TDSQL-C Cluster List Beijing 3 Other regions 12

Create Quick Check Renew More Separate keyword

Cluster ID/Name	Cluster ...	Compatible Data...	AZ	Read/Write Address	Read-Only Address	Database Proxy Address
cynosdbmysql-	Running	MySQL 5.7	Beijing Z...	(Private) :3306 (Public) Disabled	--	--

3. On the cluster management page, select the **Backup Management** tab and click **Roll Back**.

Monitoring and Alarms Account Management Database Management Database Proxy Parameter Setti

Roll Back Manual Backup Auto-Backup Settings

Data Backup List Binlog Backup List

4. On the database/table rollback page, set the rollback mode to **By backup file**. Then, select the database/table to be rolled back, rename it, and click **Roll Back**.

Roll back Databases/Tables to Original Cluster

Rollback Mode

By backup file

By time point

Backup File *

-- / 2023-08-30 00:34:34

Databases/Tables to Roll back

Databases/Tables to Roll back

Database

Table

Table name

Database name

Search

☒

cynosdbmysql

☒

test

You have selected 0 tables. L

be selected.

☒

cynosdbmysql

☒

test

Roll Back

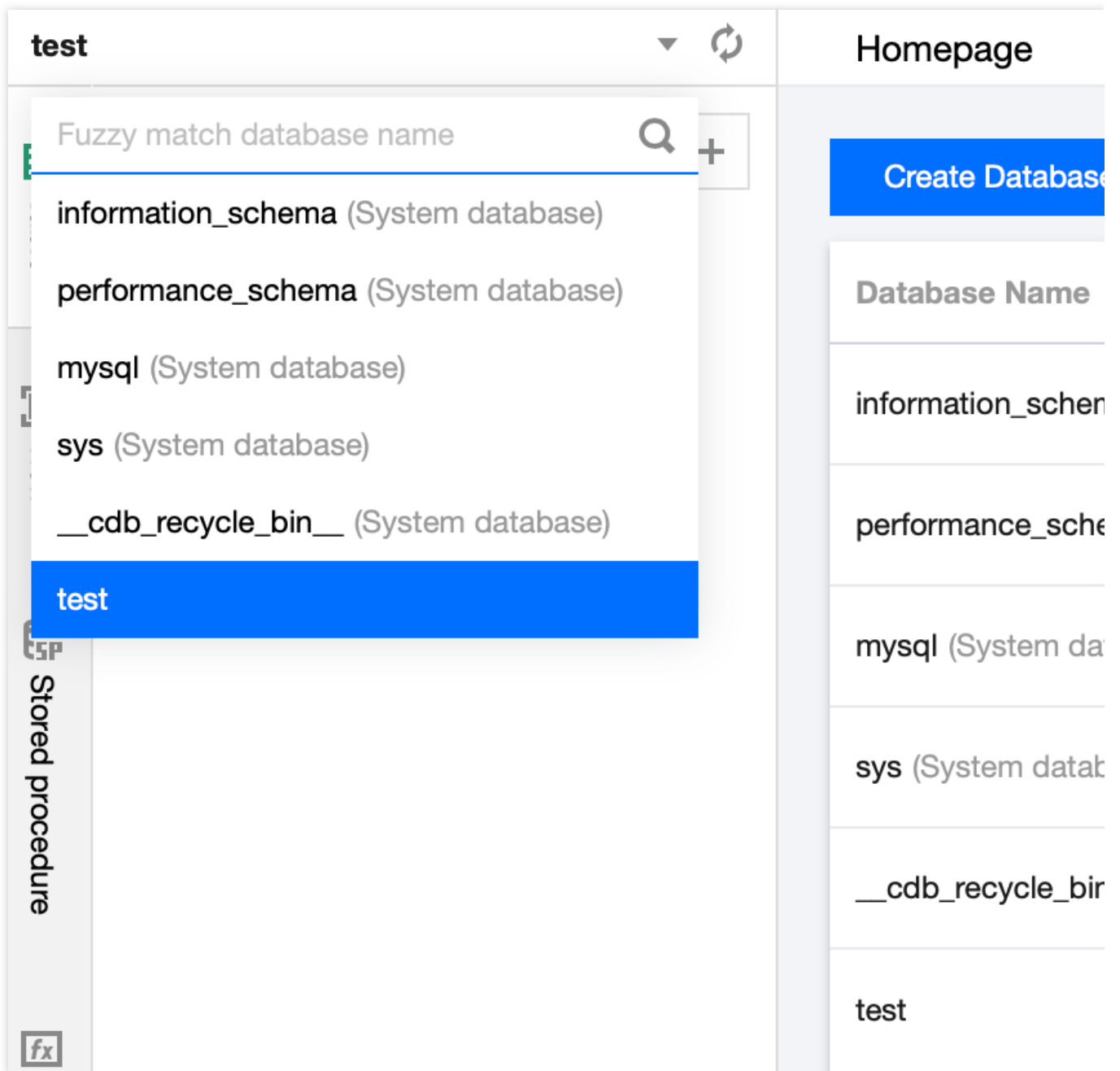
Cancel

5. In the pop-up window, confirm that everything is correct and click **OK** to initiate the rollback task.
6. After the task is submitted, you can click **View Rollback Task** or go to the task list to view the rollback progress and task details.
7. After the rollback is completed, you can see the new restored database/table in the original cluster and perform next operations.

Step 2. Compare the data

After the databases/tables are rolled back, you can log in to the cluster through DMC to compare and verify the data.

1. Log in to the [TDSQL-C for MySQL console](#).
2. In the cluster list, find the rollback cluster and click **Log In**.
3. On the login page, enter the cluster account and password, and click **Log In**.



4. After entering the DMC management page, select the rollback database from the drop-down menu on the left and find the misoperation data. Then, verify that the data has been rolled back to its original status and other data is also consistent with how it was before the misoperation.

test

Fuzzy match database name

information_schema (System database)

performance_schema (System database)

mysql (System database)

sys (System database)

__cdb_recycle_bin__ (System database)

test

Homepage

Create Database

Database Name
information_schen
performance_sche
mysql (System da
sys (System data
__cdb_recycle_bir
test

Rolling back Entire Cluster

Overview

Last updated : 2023-11-09 11:32:59

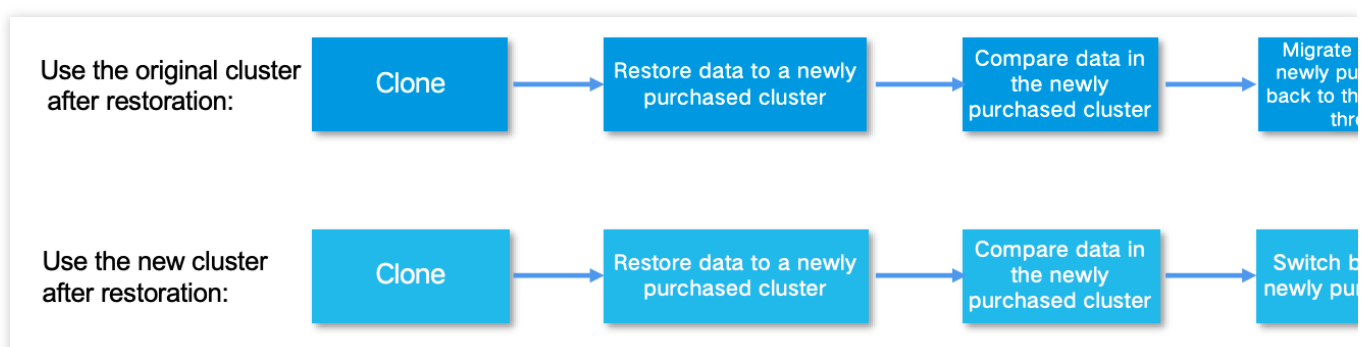
TDSQL-C for MySQL provides the clone feature to restore an entire cluster to any time point in the log backup retention period or to the backup set of the specified backup file. The clone will create a new cluster based on your choice. After the new cluster is verified, you can either migrate the data back to the original cluster through DTS or use the newly cloned cluster directly. If the original cluster is no longer needed, you can delete it. For more information, see [Migrating to TDSQL-C for MySQL](#).

TDSQL-C for MySQL supports the cluster rollback by time point and backup set. If you choose to roll back by time point, the log retention period you specify will define the selection range of time points. If you choose to roll back by backup set, the data backup retention time you specify will define the selection range of backup files. After a cluster is created, TDSQL-C for MySQL generates backup files every 6–4848 hours on a 24/24 uninterrupted basis, which has no impacts on the instance performance. You can also set the backup retention time in the console as needed. Based on this, when you perform cloning on the cluster, the available time points and backup sets will be directly displayed for your selection in the console.

Process

To roll back by either time point or backup set, you first need to roll back the full backup data to a newly purchased cluster. Once you've verified the data in the new cluster, you can migrate the rolled-back data back to the original cluster and terminate the new one. Alternatively, you can directly move your business to the new cluster and terminate the original one.

The process is as shown below:



Instructions

Cluster Rollback by Time Point

Cluster Rollback by Backup Set

Solution to High CPU Utilization

Last updated : 2024-06-07 14:22:56

Problem Description

High CPU utilization in TDSQL-C for MySQL clusters can often lead to system anomalies, such as slow responses, inability to obtain connections, and timeout. A large number of timeout retries are often the main culprits of performance "avalanches". High CPU utilization is often caused by abnormal SQL statements, and a large number of lock conflicts, lock waits, or unsubmitted transactions can also lead to high CPU utilization.

When the database performs business queries or modifies statements, the CPU first requests data blocks from the memory:

If the memory has the target data, the CPU will execute the computation task and return the result, which may involve actions requiring high CPU utilization such as sorting.

If the memory does not have the target data, the database will get the data from the disk.

The two data acquisition processes above are called logical read and physical read, respectively. Therefore, poorly performing SQL statements can easily cause the database to generate a lot of logical reads during the execution, resulting in high CPU utilization. They may also make the database generate a lot of physical reads, resulting in high IOPS and I/O latency.

Solutions

[DBbrain](#) provides users with three major features to identify and optimize the abnormal SQL statements that cause high CPU utilization:

Anomaly diagnosis: It supports 7 * 24-hour anomaly detection and diagnosis, providing real-time optimization suggestions.

Slow SQL analysis: It analyzes slow SQL statements of the current instance and provides corresponding optimization suggestions.

Audit log analysis: It performs in-depth analysis on SQL statements and provides optimization suggestions based on TencentDB audit data (full SQL).

Method 1 (recommended): Use the "exception diagnosis" feature to troubleshoot database exceptions.

The exception diagnosis feature offers proactive fault localization and optimization, requiring no database operation and maintenance experience. It addresses not only exceptions of high CPU utilization but also nearly all frequent exceptions and failures in both read/write instances and read-only instances in a cluster.

The steps are as shown in the example below:

1. log in to the [DBbrain console](#), select **Performance Optimization** from the left navigation pane, and then click the **Exception Diagnosis** tab on the top.
2. Select (enter or search for) an instance ID in the top-left corner to switch to the target instance.
3. On this page, select **Real-Time** or **Historical** and specify the time to be queried. If there are any failures within this time frame, an overview of the information can be viewed in the "Diagnosis Prompt" on the right.
4. Click **View Details** in the "Real-Time/Historical Diagnosis" or the diagnostic items in the **Diagnosis Prompt** column to enter the diagnosis details page.

Event overview: Includes the diagnosis item name, time range, risk level, duration, and overview.

Description: Includes symptom snapshots and performance trends of the exception event or health check event.

Intelligent Analysis: Analyzes the root cause of the performance exception to help you locate the specific operation.

Expert Suggestion: Provides optimization suggestions, including but not limited to SQL optimization (index and rewrite), resource configuration optimization, and parameter fine-tuning.

5. Click the **Optimization Suggestions** tab to view the optimization suggestions provided by DBbrain for the failure, such as optimization suggestions for SQL statements in this case.

Method 2. Use the "slow SQL analysis" feature to troubleshoot SQL statements that lead to high CPU utilization

1. Log in to the [DBbrain console](#), select **Diagnostic Optimization** from the left navigation pane, and click the **Slow SQL Analysis** tab on top.
2. Select (enter or search for) an instance ID in the top-left corner to switch to the target instance.
3. On the page, select the time period you wish to query. If there are slow SQL statements during this period, the SQL statistics section will display them in a bar chart, showing the times and quantities of slow SQL occurrences. Click on the bar chart, and the list below will display all the related slow SQL information (aggregated SQL templates), and the right side will display the execution time distribution of SQL during that period.
4. You can identify and filter SQL statement execution data in the SQL statement list in the following way:
 - 4.1 Sort the SQL statements by average duration (or maximum duration). Examine the top SQL statements in terms of duration. We do not recommend you sort the statements by total duration, as the data may be affected by a high number of executions.
 - 4.2 Then, check the numbers of returned rows and scanned rows.

If there is an SQL statement with the same "number of returned rows" and "number of scanned rows", it is very likely that the full table has been queried and returned.

If there are several SQL statements with a large number of scanned rows but no or few returned rows, it means that the system generated a lot of logical and physical reads. If the volume of the data to be queried is too high and memory is insufficient, the request will generate many physical I/O requests and consume lots of I/O resources. Too many logical reads will occupy too many CPU resources, resulting in high CPU utilization.

5. Click an SQL statement to view its details, resource consumption, and optimization suggestions.

Analysis page: You can view the complete SQL template, SQL samples, and optimization suggestions and descriptions. You can optimize SQL based on the expert recommendations provided by DBbrain to improve SQL performance and reduce execution time.

Statistics page: Based on the total execution time proportion, total lock wait time proportion, total rows scanned proportion, and total rows returned proportion in the statistics report, you can analyze the specific causes of the slow SQL occurrence and perform corresponding optimization.

Details page: You can view the user source, IP source, database, and other detailed information for this type of SQL.

How to Authorize Sub-Users to View Monitoring Data

Last updated : 2024-06-07 14:23:46

Operation Scenarios

By default, the root account is the owner of the resources and has access permissions to all its resources. Sub-users do not have access permissions to any resources by default and require the root account to grant them the relevant access permissions. Only then can a sub-user normally access the relevant resources. Therefore, for a sub-user to be able to log in and access the monitoring and alarm features of the TDSQL-C for MySQL cluster, the root account must first authorize the sub-user.

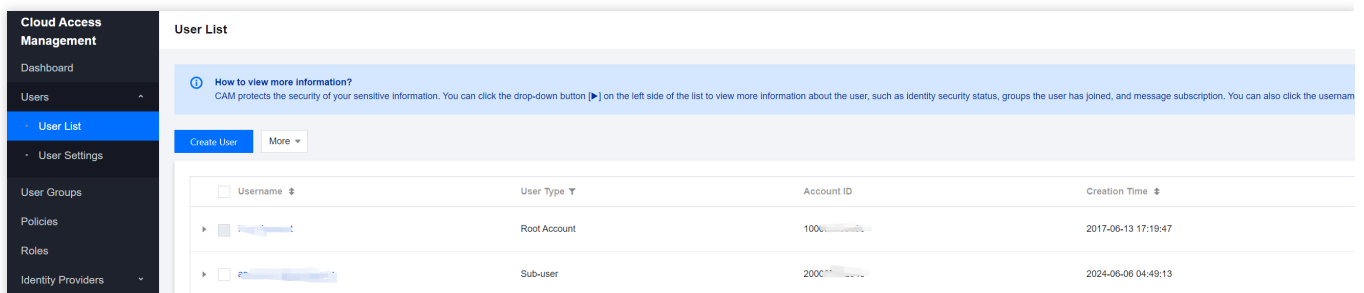
When an unauthorized sub-user account is used to log in to the console, you cannot query instance monitoring information, or access TCOP and set alarms.

Prerequisites

To authorize a sub-user, you need to log in to the CAM console with the root account. Contact the root account holder in advance to assist with the authorization operation.

Directions

1. Log in to the [CAM console](#) with the root account.
2. In the left navigation pane, choose **Users > User List**.
3. In the user list, find the target sub-user, and click **Authorize** in the **Operation** column.



4. In the pop-up window, find and select the relevant policies (see table below), and click **OK**.

Associate Policy

Select Policies (1 Total)

QcloudCynosDBFullAccess

Policy Name

Policy Type ▾

QcloudCynosDBFullAccess
Full read-write access to CynosDB

Preset Policy

1 selected

Policy Name

QcloudCynosDBFullAccess
Full read-write access to CynosDB

Support for holding shift key down for multiple selection

OK

Cancel

Policy Name	Description
QcloudCynosDBFullAccess	For full read-write access to TDSQL-C for CynosDB, the authorized sub-user can view the TDSQL-C for MySQL clusters under the root account and has full read-write access to the clusters.
QcloudCynosDBReadOnlyAccess	For read-only access to TDSQL-C for CynosDB, the authorized sub-user can view the TDSQL-C for MySQL clusters under the root account and has read-only access to the clusters.
QcloudMonitorFullAccess	Full read-write access to TCOP (Monitor), including the permission to view user groups.
QcloudMonitorReadOnlyAccess	Read-only access to TCOP (Monitor).

5. After authorization, the sub-user can log in to the [TDSQL-C for MySQL console](#) to view monitoring information and set alarms for the relevant clusters.

