

Data Transmission Service

Best Practices

Product Documentation



Copyright Notice

©2013-2019 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

Creating Two-Way Sync Data Structure

Creating Many-to-One Sync Data Structure

Creating Multi-Site Active-Active IDC Architecture

Migrating Data from Self-Built MySQL to TencentDB for MySQL Through CCN

Best Practices

Creating Two-Way Sync Data Structure

Last updated : 2022-05-17 10:30:13

Overview

DTS supports two-way data sync between two databases, which can be applied to multi-site active-active scenarios. In a two-way sync task, two one-way sync tasks are created to establish a two-way topology, and data can be written into both databases at the same time during sync.

As a two-way sync task consists of two one-way sync tasks to establish a two-way topology, restrictions on one-way sync and relevant operations must be followed. For more information, see the appropriate sync scenario in [Data Sync](#).

Notes

- During full data sync, DTS consumes certain source database resources, which may increase the load and pressure of the source database. If your database configuration is low, we recommend you sync the data during off-peak hours.
- To avoid duplicate data, make sure that the tables to be synced have a primary key or non-null unique key.
- You should plan the data in advance. The two source databases are responsible for updating (adding, deleting, and modifying) data with different primary keys so as to avoid problems such as primary key conflict and mutual overwriting of data with the same primary key. If there are duplicate primary keys in the two source databases for business reasons, select an appropriate conflict resolution method as instructed in [Recommended Configurations for Typical Use Cases](#) to make the sync behavior and data meet the expectations.

Application Restrictions

- DDL statements can be executed in at most one direction during two-way sync, as the sync linkage should not form a ring (you can run DDL statements in either the forward or reverse direction).
- Currently, a two-way sync task can be created between two MySQL databases, two TDSQL-C for MySQL databases, or one MySQL database and one TDSQL-C for MySQL database.

Recommended Configurations for Typical Use Cases

A two-way sync task consists of two one-way sync tasks to establish a two-way topology. The creation steps for each one-way sync task are similar to those for a general one-way sync task. They differ only in the following sync option settings:

Sync Option Settings Difference

1 Set source and target databases >
2 Set sync options and objects >
3 Verify task

Data Initialization Option

Initialization Type: Structure Initialization Full data initialization

If Target Already Exists *: Precheck and report error Ignore and execute

Data Sync Option

Conflict Resolution Method *: Report Ignore Overwrite

SQL Type

DML *: Insert Update Delete

DDL: DDL

Custom DDL *:

The following configurations are recommended for typical use cases for your reference. Example: a two-way sync task consisting of two one-way sync tasks (1 and 2) needs to be created between databases A and B: A > B (task 1) and B > A (task 2).

Scenario	Time Requirements	Sync Task	Initialization Type	If Target Already Exists	Conflict Resolution Method	SQL Type
Scenario 1: database A has database/table structures and data, and database B is empty	Task 2 can be created only after task 1 enters the "incremental sync" phase	Task 1: forward sync	Structure initialization/full data initialization	Precheck and report error	Select an option as needed. • Example: if a primary key conflict occurs, and you want the content of database A to prevail,	<ul style="list-style-type: none"> • Select DDL in at most one task. • For operation types other than DDL, keep them consistent
		Task 2: reverse sync	Do not select	Ignore and execute		
Scenario 2: database A	None	Task 1: forward	Full data initialization	Ignore and		

has database/table structures and data, and database B has only database/table structures but no data		sync		execute	you need to select **Overwrite** for task 1 and **Ignore** or **Report** for task 2. • The conflict resolution method takes effect only for the data with primary key conflict.	between the two tasks.
		Task 2: reverse sync	Do not select	Ignore and execute		
Scenario 3: both databases A and B have database/table structures and data	None	Task 1: forward sync	Full data initialization	Ignore and execute		
		Task 2: reverse sync	Full data initialization	Ignore and execute		

Directions

This document uses MySQL two-way sync (where database A has database/table structures and data, and database B has only database/table structures but no data) as an example. The two-way sync operations for other databases are similar.

Creating sync task 1: forward sync (database A > database B)

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

Parameter	Description
Billing Mode	Monthly subscription and pay-as-you-go billing modes are supported. Currently, the data sync feature is free of charge, and you will receive notifications by email and Message Center one month before the billing officially starts.
Source Database Type	Select MySQL (including TencentDB for MySQL and self-built MySQL).
Source Database Region	Select the source database A region.

Parameter	Description
Target Database Type	Select MySQL (including TencentDB for MySQL and self-built MySQL).
Target Database Region	Select the target database B region.
Sync Task Specification	Currently, only the Standard Edition is supported

- After successful purchase, return to the [data sync list](#), and you can see the newly created data sync task. You need to configure it before you can use it.
- In the data sync list, click **Configure** in the **Operation** column to enter the sync task configuration page.
- On the sync task configuration page, configure the source and target databases and their accounts and passwords, test the connectivity, and click **Next**.

Category	Parameter	Description
Task Configuration	Task Name	DTS will automatically generate a task name, which is customizable.
	Running Mode	Immediate execution and scheduled execution are supported.
Source Database Settings	Source Database Type	Select the TencentDB instance type selected during purchase, which cannot be changed once configured.
	Source Database Region	Select the TencentDB instance A region selected during purchase, which cannot be changed once configured.
	Service Provider	Others (including TencentDB for MySQL and self-built MySQL), AWS, and Alibaba Cloud are supported.

Category	Parameter	Description
	Access Type	<p>If Other Cloud Vendors is selected as Service Provider, the access type can be public network; if Others is selected as Service Provider, you need to select an access type according to the database deployment conditions.</p> <ul style="list-style-type: none"> Public Network: self-built database connected through a public IP. Self-Build on CVM: self-built database on CVM. Direct Connect/VPN Access: self-built database connected through a Direct Connect/VPN gateway. VPC: self-built database connected through a VPC. Database: TencentDB database. CCN: self-built database connected through CCN.
Target Database Settings	Target Database Type	Select the target database type, which cannot be changed once configured.
	Target Database Region	Select the target database B region, which cannot be changed once configured.
	Access Type	Select the access type of the target database B.

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**.

Category	Parameter	Description
Data Initialization Option	Initialization Type	<ul style="list-style-type: none"> Structure initialization: table structures in the source database will be initialized into the target database before the sync task runs. Full data initialization: data in the source database will be initialized into the target database before the sync task runs. <p>In this document, select Full data initialization.</p>
	If Target Already Exists	<ul style="list-style-type: none"> 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. Ignore and execute: full and incremental data will be directly added to tables in the target database. <p>In this document, select Ignore and execute.</p>

Category	Parameter	Description
Data Sync Option	Conflict Resolution Method	<ul style="list-style-type: none"> Report: if a primary key conflict is found during data sync, an error will be reported, and the data sync task will be paused. Ignore: if a primary key conflict is found during data sync, the primary key record in the target database will be retained. 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. Select an option as needed.
	SQL Type	Supported operations: INSERT, UPDATE, DELETE, and DDL. In two-way sync, you can select DDL in at most one task. In this document, select DDL in task 1 but not task 2.
Sync Object Option	Database and Table Objects of Source Database	Select the objects to be synced. You can select databases, tables, and views.
	Selected Object	It displays the selected sync objects, and database/table mapping is supported.

6. On the task verification page, the system will check the DDL configuration first and then check the source and target database parameters. After all check items are passed, click **Start Task**.

Note :

- If the verification failed, fix the problem as instructed in [Fix for Verification Failure](#) and initiate the verification task again.
- If an alarm is displayed in the verification result, it will not affect the task start, but we recommend you click **View Details** to get the suggestions for adjustment.

- DDL check
- Source and target database parameter check

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

Creating sync task 2: reverse sync (database B > database A)

The operations of the two sync tasks are basically the same. The following only describes their differences:

1. Set the sync source and target databases.

Swap the settings of the source and target databases in task 1.

2. Set the sync options and objects.

- Initialization Type: do not select.
- If Target Already Exists: select **Ignore and execute**.
- Conflict Resolution Method: select the same option as selected in task 1.
- SQL Type: in two-way sync, you can select DDL in at most one task. In this document, select DDL in task 1 but not task 2.

Stopping sync task

If you no longer need a sync task, you can select **More > Stop** in the **Operation** column to stop it.

Creating Many-to-One Sync Data Structure

Last updated : 2022-05-25 14:14:54

Overview

Many-to-One sync is to sync the content in multiple source databases to one target database. If you use a single database, you may often need to shard the data due to high load or region issues, but storing the databases/tables of the same type in many databases makes data query inconvenient. The many-to-one sync feature can easily solve this problem.

As a many-to-one sync task consists of multiple one-way sync tasks to establish a many-to-one topology, restrictions on one-way sync and relevant operations must be followed. For more information, see the appropriate sync scenario in [Data Sync](#).

Notes

- During full data sync, DTS consumes certain source database resources, which may increase the load and pressure of the source database. If your database configuration is low, we recommend you sync the data during off-peak hours.
- To avoid duplicate data, make sure that the tables to be synced have a primary key or non-null unique key.
- You should plan the data in advance. Each source database is responsible for updating (adding, deleting, and modifying) data with different primary keys so as to avoid problems such as primary key conflict and mutual overwriting of data with the same primary key. If there are duplicate primary keys in multiple source databases for business reasons, select an appropriate conflict resolution method as instructed in [Recommended Configuration for Typical Use Case](#) to make the sync behavior and data meet the expectations.

Application Restrictions

- DDL statements in the configurations of multiple sync tasks should not form a ring.
- Currently, a many-to-one sync task can be created between two MySQL databases, two TDSQL-C for MySQL databases, or one MySQL database and one TDSQL-C for MySQL database.

DDL Configuration Principles

- DDL statements in the configurations of multiple sync tasks should not form a ring; otherwise, they will loop in the system, causing errors.
- The same table object in the target database cannot receive DDL sync from multiple source databases; otherwise, such DDL statements may conflict with each other in the target database, causing errors.
 - In many-to-one sync that combines multiple tables with the same name into one, you can select DDL in only one sync task.
 - In other types of many-to-one sync tasks (such as a task that combines multiple tables with different names into one database), you can select DDL in each task. In this case, select an appropriate DDL sync policy based on the actual conditions.
- During verification, the sync system will judge whether the sync task being created will cause a DDL loop or conflict based on all your other sync tasks and provide prompts for your reference.

Recommended Configurations for Typical Use Cases

A many-to-one sync task consists of multiple one-way sync tasks to establish a many-to-one topology. The creation steps for each one-way sync task are similar to those for a general one-way sync task. They differ only in the following sync option settings:

The following configurations are recommended for typical use cases for your reference.

Example: a sync task among databases A, B, and C needs to be created, where databases A and B have tables with the same name that need to be synced to database C, task 1 is sync from A to C, and task 2 is sync from B to C. To sync data from more source databases to the target database, simply add sync tasks by referring to task 2.

Scenario	Time Requirements	Sync Task	Initialization Type	If Target Already Exists	Conflict Resolution Method	SQL Type
Scenario 1: databases A and B have database/table structures and data, and database C is empty	Task 2 can be started only after task 1 enters the "incremental sync" phase	Task 1	Structure initialization/full data initialization	Ignore and execute	Select an option as needed. <ul style="list-style-type: none"> Example: if a primary key conflict occurs, and you want the content of database A to prevail, you need to select **Overwrite** for task 1 and 	<ul style="list-style-type: none"> Select DDL in at most one task. For operation types other than DDL, keep them consistent between the other multiple tasks.
		Task 2	Full data initialization	Ignore and execute		
Scenario 2: databases A and B have database/table structures and	None	Task 1	Full data initialization	Ignore and execute		
		Task	Same as task	Same as		

data, and database C has only database/table structures but no data		2	1	task 1	**Ignore** or **Report** for task 2. • The conflict resolution method takes effect only for the data with primary key conflict.
Scenario 3: databases A, B, and C all have database/table structures and data	None	Task 1	Full data initialization	Ignore and execute	
		Task 2	Same as task 1	Same as task 1	

Directions

The following uses MySQL two-to-one sync (databases A and B have database/table structures and data, and database C is empty) as an example. The many-to-one sync operations for other databases are similar.

Creating sync task 1 (database A > database C)

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

Parameter	Description
Billing Mode	Monthly subscription and pay-as-you-go billing modes are supported. Currently, the data sync feature is free of charge, and you will receive notifications by email and Message Center one month before the billing officially starts.
Source Database Type	Select MySQL (including TencentDB for MySQL and self-built MySQL).
Source Database Region	Select the source database A region.
Target Database Type	Select MySQL (including TencentDB for MySQL and self-built MySQL).
Target Database Region	Select the target database C region.

Parameter	Description
Sync Task Specification	Currently, only the Standard Edition is supported

- After successful purchase, return to the [data sync list](#), and you can see the newly created data sync task. You need to configure it before you can use it.
- In the data sync list, click **Configure** in the **Operation** column to enter the sync task configuration page.
- On the sync task configuration page, configure the source and target databases and their accounts and passwords, test the connectivity, and click **Next**.

Category	Parameter	Description
Task Configuration	Task Name	DTS will automatically generate a task name, which is customizable.
	Running Mode	Immediate execution and scheduled execution are supported.
Source Database Settings	Source Database Type	Select the TencentDB instance type selected during purchase, which cannot be changed once configured.
	Source Database Region	Select the TencentDB instance A region selected during purchase, which cannot be changed once configured.
	Service Provider	Others (including TencentDB for MySQL and self-built MySQL), AWS, and Alibaba Cloud are supported.
	Access Type	<p>If Other Cloud Vendors is selected as Service Provider, the access type can be public network; if Others is selected as Service Provider, you need to select an access type according to the database deployment conditions.</p> <ul style="list-style-type: none"> Public Network: self-built database connected through a public IP. Self-Build on CVM: self-built database on CVM. Direct Connect/VPN Access: self-built database connected through a Direct Connect/VPN gateway. VPC: self-built database connected through a VPC. Database: TencentDB database. CCN: self-built database connected through CCN.
Target Database Settings	Target Database Type	Select the target database type, which cannot be changed once configured.

Category	Parameter	Description
	Target Database Region	Select the target database C region, which cannot be changed once configured.
	Access Type	Select the access type of the target database C.

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**.

Category	Parameter	Description
Data Initialization Option	Initialization Type	<ul style="list-style-type: none"> Structure initialization: table structures in the source database will be initialized into the target database before the sync task runs. Full data initialization: data in the source database will be initialized into the target database before the sync task runs. In this document, select **Structure initialization/Full data initialization** .
	If Target Already Exists	<ul style="list-style-type: none"> 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. Ignore and execute: full and incremental data will be directly added to tables in the target database. In this document, select **Ignore and execute** .
Data Sync Option	Conflict Resolution Method	<ul style="list-style-type: none"> Report: if a primary key conflict is found during data sync, an error will be reported, and the data sync task will be paused. Ignore: if a primary key conflict is found during data sync, the primary key record in the target database will be retained. 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. Select an option as needed.
	SQL Type	Supported operations: INSERT, UPDATE, DELETE, and DDL. In many-to-one sync, you can select DDL in at most one task. In this document, select DDL in task 1 but not other tasks.
Sync Object Option	Database and Table Objects of Source Database	Select the objects to be synced. You can select databases, tables, and views.

Category	Parameter	Description
	Selected Object	It displays the selected sync objects, and database/table mapping is supported.

6. On the task verification page, the system will check the DDL configuration first and then check the source and target database parameters. After all check items are passed, click **Start Task**.

Note :

- If the verification failed, fix the problem as instructed in [Fix for Verification Failure](#) and initiate the verification task again.
- If an alarm is displayed in the verification result, it will not affect the task start, but we recommend you click **View Details** to get the suggestions for adjustment.

- DDL check
- Source and target database parameter check

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

Creating sync task 2 (database B > database C)

Configure sync task 2 after the previous sync task enters the **incremental sync** phase.

The operations of tasks 1 and 2 are basically the same. The following only describes their differences:

1. Set the sync source and target databases.
 - Enter the information of databases A and B in the source and target database settings respectively.
2. Set the sync options and objects.
 - Initialization Type: select **Full data initialization** only but not **Structure Initialization**.
 - If Target Already Exists: select **Ignore and execute**.
 - Conflict Resolution Method: select an option as needed.
 - SQL Type: do not select DDL. In many-to-one sync, you can select DDL in at most one task. In this document, select DDL in task 1 but not other tasks.

Stopping sync task

If you no longer need a sync task, you can select **More > Stop** in the **Operation** column to stop it.

Creating Multi-Site Active-Active IDC Architecture

Last updated : 2021-12-27 11:46:58

Overview

The multi-site active-active IDC architecture refers to multiple IDCs that are deployed in different regions and provide service concurrently. Data can be synced among them in real time. If a disaster occurs in an IDC, its traffic can be routed to other IDCs to implement quick cross-region failover and guarantee business continuity.

The multi-site active-active IDC architecture is implemented by creating multiple two-way sync tasks, each of which consists of two one-way sync tasks. Therefore, restrictions on one-way sync and relevant operations must be followed. For more information, see the appropriate sync scenario in [Data Sync](#).

Notes

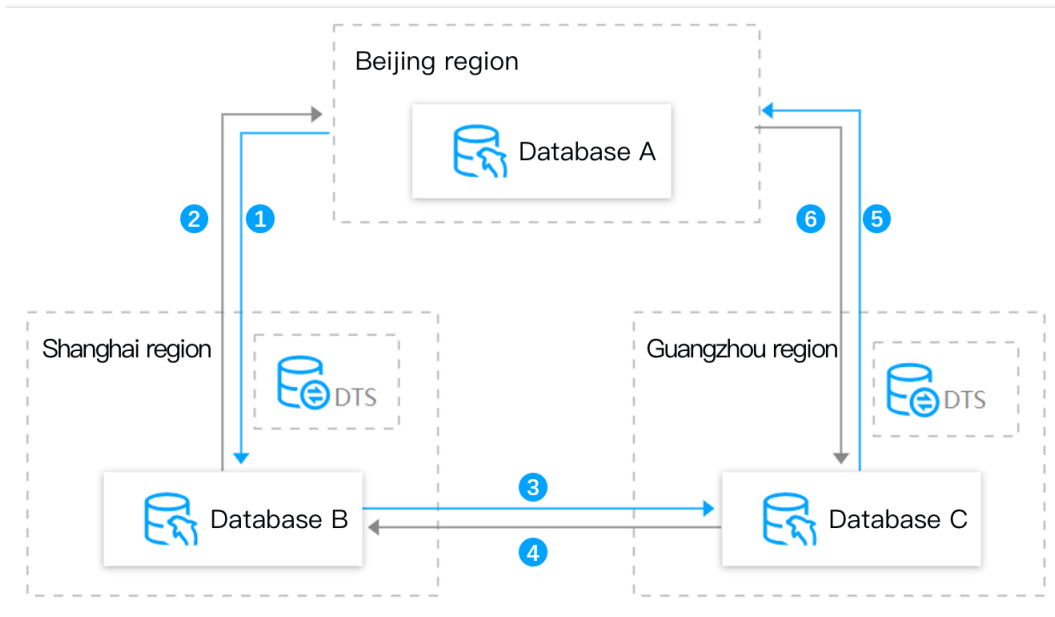
- During full data sync, DTS consumes certain source database resources, which may increase the load and pressure of the source database. If your database configuration is low, we recommend you sync the data during off-peak hours.
- To avoid duplicate data, make sure that the tables to be synced have a primary key or non-null unique key.
- You should plan the data in advance. Each IDC is responsible for updating (adding, deleting, and modifying) data with different primary keys so as to avoid problems such as primary key conflict and mutual overwriting of data with the same primary key. If there are duplicate primary keys in multiple source databases for business reasons, select an appropriate conflict resolution method to make the sync behavior and data meet the expectations.

Application Restrictions

- DDL statements in the configurations of multiple sync tasks should not form a ring.
- Currently, a two-way sync task can be created between two MySQL databases, two TDSQL-C for MySQL databases, or one MySQL database and one TDSQL-C for MySQL database.

DDL Configuration Principles

This document uses a specific scenario to describe how to configure DDL statements for easier understanding. For example, in a multi-site active-active-active IDC architecture, three two-way sync tasks are created among databases A (Beijing region), B (Shanghai region), and C (Guangzhou region): A <-> B, B <-> C, and C <-> A.



- DDL statements in the configurations of multiple sync tasks should not form a ring; otherwise, they will loop in the system, causing errors.
For example, among the three sync tasks (1, 3, and 5) marked by blue lines in the following figure, you can select DDL in up to two of them, and if you select three, a ring will be formed.
- The same table object cannot receive DDL sync from multiple IDCs; otherwise, such DDL statements may conflict with each other in the target database, causing errors.
For example, databases A and C have tables with the same name to be synced to database B. Then, you can select DDL in only one task between tasks 1 and 4.
- During verification, the sync system will judge whether the sync task being created will cause a DDL loop or conflict based on all your other sync tasks and provide prompts for your reference.

Recommended Configurations for Typical Use Cases

The multi-site active-active IDC architecture is implemented by creating multiple two-way sync tasks, each of which consists of two one-way sync tasks. Therefore, the operation steps for each sync task in such architecture are basically the same as those for a general one-way sync task. They differ only in the following configurations:

Sync Option Settings Difference

1 Set source and target databases >
 2 Set sync options and objects >
 3 Verify task

Data Initialization Option

Initialization Type: Structure Initialization Full data initialization

If Target Already Exists *: Precheck and report error Ignore and execute

Data Sync Option

Conflict Resolution Method *: Report Ignore Overwrite

SQL Type

DML *: Insert Update Delete

DDL: DDL

Custom DDL *:

This document recommends the following configuration for a typical multi-site active-active IDC architecture for your reference.

For example, in a multi-site active-active-active IDC architecture, three two-way sync tasks are created among databases A (Beijing region), B (Shanghai region), and C (Guangzhou region): A <-> B (tasks 1 and 2), B <-> C (tasks 3 and 4), and C <-> A (tasks 5 and 6).

Scenario	Time Requirements	Sync Task	Initialization Type	If Target Already Exists	Conflict Resolution Method	SQL Type
Scenario 1: database A has database/table structures and data, and databases B and C are empty	Task 2 can be created only after task 1 enters the "incremental sync" phase	Task 1	Structure initialization/full data initialization	Precheck and report error	Select an option as needed. The conflict resolution method takes effect only for the data with primary key conflict.	Select DDL statements according to the configuration principles. For other operation types, we recommend you keep them consistent
		Task 2	Do not select	Ignore and execute		
	Task 3	Structure initialization/full data initialization	Precheck and report error			

	"incremental sync" phase	Task 4	Do not select	Ignore and execute	among all sync tasks.
	Task 6 can be created only after task 5 enters the "incremental sync" phase	Task 5	Structure initialization/full data initialization	Precheck and report error	
		Task 6	Do not select	Ignore and execute	
Scenario 2: databases A, B, and C all have database/table structures and data	None	Tasks 1-6	Full data initialization	Ignore and execute	

Directions

Creating a multi-site active-active IDC architecture is to create multiple two-way sync tasks. For detailed directions, see [Creating Two-Way Sync Data Structure](#).

Migrating Data from Self-Built MySQL to TencentDB for MySQL Through CCN

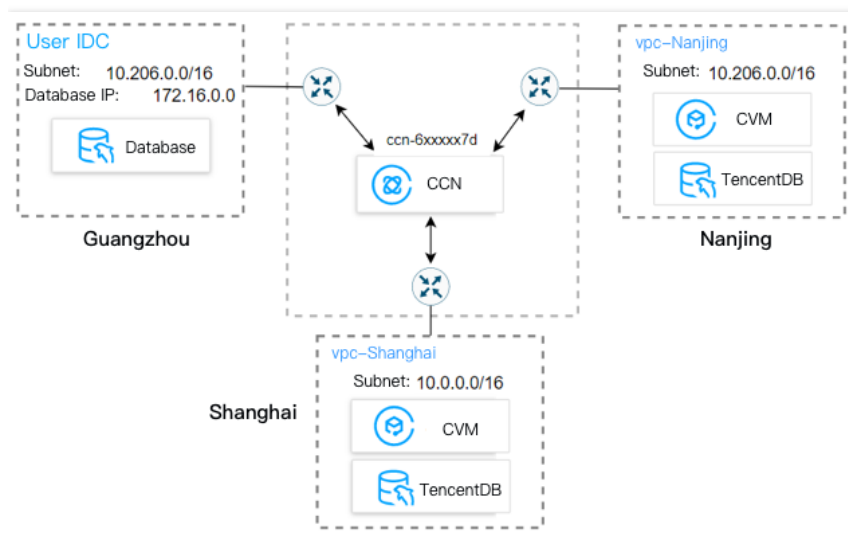
Last updated : 2021-12-28 16:49:47

Overview

This document describes how to use the data migration feature of DTS to migrate data from a self-built database to a TencentDB database through CCN.

CCN can interconnect a VPC with another VPC or a local IDC. To use CCN access, you must establish cross-VPC and VPC-IDC interconnections through CCN in advance.

In this scenario, you have used CCN to interconnect the following three networks: vpc-Guangzhou, vpc-Nanjing, and vpc-Shanghai, and plan to migrate a MySQL database in the Guangzhou region to the target database through DTS. vpc-Nanjing or vpc-Shanghai is selected as the VPC associated with CCN.



Notes

- During full data migration, DTS consumes certain source database resources, which may increase the load and pressure of the source database. If your database configuration is low, we recommend you migrate the data during off-peak hours.
- Full migration is implemented with tables locked, during which write operations will be blocked for seconds.

Prerequisites

- You have created a [TencentDB for MySQL](#) instance.
- The source and target databases must meet the requirements for the migration feature and version as instructed in [Databases Supported by Data Migration](#).
- You have completed all [preparations](#).
- The source database must have the following permissions:

- Migration of the entire instance:

```
CREATE USER 'migration account'@'%' IDENTIFIED BY 'migration password';
GRANT RELOAD, LOCK TABLES, REPLICATION CLIENT, REPLICATION SLAVE, SHOW DATABASES,
SHOW VIEW, PROCESS ON *.* TO 'migration account'@'%';
GRANT ALL PRIVILEGES ON `__tencentdb__`.* TO 'migration account'@'%';
GRANT SELECT ON *.* TO 'migration account';
```

- Migration of specified objects:

```
CREATE USER 'migration account'@'%' IDENTIFIED BY 'migration password';
GRANT RELOAD, LOCK TABLES, REPLICATION CLIENT, REPLICATION SLAVE, SHOW DATABASES,
SHOW VIEW, PROCESS ON *.* TO 'migration account'@'%';
GRANT ALL PRIVILEGES ON `__tencentdb__`.* TO 'migration account'@'%';
GRANT SELECT ON `mysql`.* TO 'migration account'@'%';
GRANT SELECT ON database to be migrated.* TO 'migration account';
```

- Permissions required of 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.

Application Restrictions

- Only basic tables and views can be migrated, while objects such as functions, triggers, and stored procedures cannot.
- System databases/tables and user information including `information_schema`, `sys`, `performance_schema`, `__cdb_recycle_bin__`, `__recycle_bin__`, `__tencentdb__`, and `mysql` cannot be migrated. After the migration is completed, if you want a view, stored procedure, or function in the target database to be called, you must grant the caller the read/write permissions.
- When a view is exported, DTS will check whether `user1` corresponding to `DEFINER` (`[DEFINER = user1]`) in the source database is the same as `user2` in the migration target, and if not, DTS will change the `SQL SECURITY` attribute of `user1` in the target database from `DEFINER` to `INVOKER` (`[INVOKER =`

`user1]`), and set the `DEFINER` in the target database to `user2` of the migration target (`[DEFINER = migration target user2]`).

- If the source MySQL database is a non-GTID database, DTS doesn't support HA switch for it. If it is switched, DTS incremental sync may be interrupted.
- Only data with the following three database engines can be migrated: InnoDB, MySIAM, and TokuDB. Tables with other engines will be skipped during migration by default.
- Correlated data objects must be migrated together; otherwise, migration will fail. Common correlations include table reference by views, view reference by views, view/table reference by stored procedures/functions/triggers, and tables correlated through primary/foreign keys.
- During incremental migration, if the source database has distributed transactions or generates binlog statements in the `STATEMENT` format, the migration will fail.
- In migration without locks (the source database is Alibaba Cloud ApsaraDB RDS for MySQL 5.6, Alibaba Cloud PolarDB for MySQL 5.6, or Amazon RDS for MySQL, and the target database is TencentDB for MySQL), DDL operations are not supported during full migration.

Operation Restrictions

- During migration, do not perform the following operations; otherwise, the migration task will fail:
 - Do not modify or delete user information (including username, password, and permissions) in the source and target databases and port numbers.
 - Do not run distributed transactions in the source database.
 - Do not write binlog data in the `STATEMENT` format into the source database.
 - Do not clear binlogs in the source database.
 - Do not run DDL operations of changing the database/table structure during database/table structure migration or full migration.
 - Do not delete the system table `__tencentdb__` during incremental migration.
- If you only perform full data migration, do not write new data into the source database during migration; otherwise, the data in the source and target databases will be inconsistent. In scenarios with data writes, to ensure the data consistency in real time, we recommend you select full + incremental data migration.

Supported SQL Operations

Operation Type	Supported SQL Operations
DML	INSERT, UPDATE, DELETE, and REPLACE

Operation Type	Supported SQL Operations
DDL	TABLE: CREATE TABLE, ALTER TABLE, DROP TABLE, TRUNCATE TABLE, and RENAME TABLE VIEW: CREATE VIEW and DROP VIEW INDEX: CREATE INDEX and DROP INDEX DATABASE: CREATE DATABASE, ALTER DATABASE, and DROP DATABASE

Environment Requirements

Note :

The system will automatically check the following environment requirements before starting a migration task and report an error if a requirement is not met. If you can identify the failed check item, fix it as instructed in [Requirements for Check Items](#); otherwise, wait for the system verification to complete and fix the problem according to the error message.

Type	Environment Requirement
Requirements for source database	<ul style="list-style-type: none"> • The source and target databases can be connected. • The server where the source database resides must have enough outbound bandwidth; otherwise, the migration speed will be affected. • Requirements for the database parameters: <ul style="list-style-type: none"> ◦ The <code>`server_id`</code> parameter in the source database must be set manually and cannot be 0. ◦ <code>`row_format`</code> for the source databases/tables cannot be set to <code>`FIXED`</code>. ◦ The values of the <code>`lower_case_table_names`</code> variable in both the source and target databases must be the same. ◦ The <code>`connect_timeout`</code> variable in the source database must be greater than or equal to 10. ◦ We recommend you enable <code>`skip-name-resolve`</code> to reduce the possibility of connection timeout. • Requirements for binlog parameters: <ul style="list-style-type: none"> ◦ The <code>`log_bin`</code> variable in the source database must be set to <code>`ON`</code>. ◦ The <code>`binlog_format`</code> variable in the source database must be set to <code>`ROW`</code>. ◦ The <code>`binlog_row_image`</code> variable in the source database must be set to <code>`FULL`</code>. ◦ On MySQL 5.6 or above, if the <code>`gtid_mode`</code> variable is not <code>`ON`</code>, an alarm will be triggered. We recommend you enable <code>`gtid_mode`</code>. ◦ You cannot set filter conditions with <code>`do_db`</code> and <code>`ignore_db`</code>.

	<ul style="list-style-type: none"> ◦ If the source database is a slave database, the <code>`log_slave_updates`</code> variable must be set to <code>`ON`</code>. • Foreign key dependency: <ul style="list-style-type: none"> ◦ Foreign key dependency can be set to only one of the following three types: <code>`NO ACTION`</code>, <code>`RESTRICT`</code>, and <code>`CASCADE`</code>. ◦ During partial table migration, tables with foreign key dependency must be migrated. • The migration precision of DTS for data in <code>`FLOAT`</code> type is 38 digits, and for data in <code>`DOUBLE`</code> type is 308 digits. You should check whether this meets your requirements.
Requirements for the target database	<ul style="list-style-type: none"> • The target database version must be above or equal to the source database version. • The size of the target database space must be at least 1.2 times that of the databases/tables to be migrated in the source database. (Full data migration will execute INSERT operations concurrently, causing some tables in the target database to generate data fragments. Therefore, after full migration is completed, the size of the tables in the target database may be larger than that in the source database.) • The target database cannot have migration objects such as tables and views with the same name as those in the source database. • The <code>`max_allowed_packet`</code> parameter of the target database must be set to 4 MB or above.
Other requirements	The environment variable <code>`innodb_stats_on_metadata`</code> must be set to <code>`OFF`</code> .

Directions

Configuring network interconnection through CCN

Establish interconnections as instructed in [Network Instance Interconnection in One Account](#).

Note :

CCN only provides bandwidth below 10 Kbps between all regions free of charge. However, DTS requires a higher bandwidth. Therefore, bandwidth configuration in the link is required.

Configuring DTS migration task

1. Log in to the [DTS console](#), select **Data Migration** on the left sidebar, and click **Create Migration Task** to enter the **Create Migration Task** page.

- On the **Create Migration Task** page, select the region of the target database and click **Free Trial**. Currently, the DTS data migration feature is free of charge.
- On the **Set source and target databases** page, configure the task, source database, and target database settings. After the source and target databases pass the connectivity test, click **Create**.

Note :

If the connectivity test fails, troubleshoot and fix the problem as prompted and as instructed in [Troubleshooting Guide](#) and try again.

Setting Type	Configuration Item	Description
Task Configuration	Task Name	Set a meaningful name for easy task identification.
	Running Mode	Immediate execution: the task will be started immediately after the task verification is passed. Scheduled execution: you need to configure a task execution time and the task will be started automatically then.
	Tag	Tags are used to manage resources by category in different dimensions. If the existing tags do not meet your requirements, go to the console to create more.
Source Database Settings	Source Database Type	Select **MySQL** .
	Service Provider	Select **Others** .
	Access Type	Select **CCN** . For more information on access types, see Overview .
	Host Address	IP address or domain name for accessing the source MySQL database.
	Port	Port for accessing the source MySQL database.
	Account	Account of the source MySQL database, which must have the required permissions.
	Password	Password of the source MySQL database.

Setting Type	Configuration Item	Description
	VPC-Based CCN Instance	Only VPC-based CCN instance is supported. You need to confirm the network type associated with CCN.
	CCN-Associated VPC	To ensure the network connectivity, you must check whether the following key requirements are met: The selected CCN-associated VPC cannot be in the same region as the host address of the source database. If the source database is MySQL in a self-built IDC, this restriction can be ignored. The selected CCN-associated VPC cannot be in the same VPC as the host address of the source database. If the source database is MySQL in a self-built IDC, the selected VPC cannot be the VPC of the Direct Connect gateway associated with the self-built IDC.
Target Database Settings	Target Database Type	Select **MySQL** .
	Access Type	Select **Database** .
	Region	Select the region selected in the previous step.
	Database Instance	Select the instance ID of the target TencentDB database.
	Account.	Account of the target TencentDB database, which must have the required permissions.
	Password	Password of the target TencentDB database.

4. On the **Set migration options and select migration objects** page, configure the migration type and objects and click **Save**.

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

Configuration Item	Description
Migration Type	<p>Select a type based on your scenario.</p> <ul style="list-style-type: none"> Structural migration: structured data such as databases and tables in the database will be migrated. Full migration: the entire database will be migrated. Full + incremental migration: the entire database and subsequent incremental data will be migrated. If there are data writes during migration, and you want to smoothly migrate the data in a non-stop manner, select this option.
Migration Object	<ul style="list-style-type: none"> Entire instance: migrate the entire database instance excluding the system databases such as `information_schema`, `mysql`, `performance_schema`, and `sys`. Specified objects: migrate specified objects.
Specify object	Select the objects to be migrated in **Source Database Object** and move them to the **Selected Object** box.

← Modify Migration Task

✔ **Set source and target databases** >
 2 **Set migration options and select migration objects** >
 3 **Verify task**

Migration Type * Structural migration Full migration **Full + Incremental migration**

Data Consistency Check ⓘ * Full check No Check

If the source instance is read-only, data consistency check will be skipped after data migration. For details, see [Migration FAQs](#).

Migration Object ⓘ * Entire instance **Specify object**

ⓘ Up to 1000 search results of source database objects are displayed by default. To view more objects, please click the "More" button or specify the object name for targeted search.

Source Database Object

Search database name, supporting fuzzy match

1 database in total, with 1 displayed More

- ▾ db-dst
 - ▶ Tables
 - ▶ Views

Refresh Select All Clear

Selected Object ⓘ Batch Rename ⓘ

Globally search for original object names, with fuzzy match supported

- ▶ db-dst

Unfold all Fold all Select All Clear

ⓘ For migration notes, please see [Migration FAQs](#).

Previous
Save

5. On the task verification page, verify the task. After the verification is passed, click **Start Task**.

If the verification failed, fix the problem as instructed in [Fix for Verification Failure](#) and initiate the verification task again.

- Failed: it indicates that a check item failed and the task is blocked. 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, and 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.

Modify Migration Task

Set source and target databases > Set migration options and select migration objects > **Verify task**

Task ID / Name	Running Mode	Source Database Type	Target Database Type	Source Access Type	Target Access Type	Address
dts-5609qg test-dtsq	Immediate execution	MySQL	MySQL	Database	Database	Source: cdb-regmpawx (trueou-dst1) Target: cdb-1mdd8uqj (trueou-dst2)

Migration Type: Full + Incremental migration

Migration Object: Specify object (🔍)
 Unfold all / Fold all
 db-dst

Create Verification Task

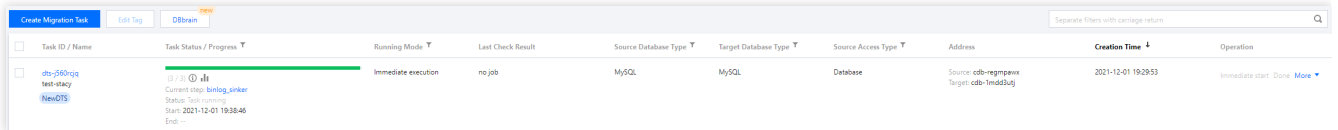
Query Verification Result

connect db check	Passed
necessary check	Passed
version check	Passed
source instance privilege check	Alarm View Details
simple instance param check	Passed
target instance privilege check	Passed
check if target instance has conflict content	Passed
check if there's enough space in target instance	Passed
source instance binlog param check	Passed
foreign key constraint check	Passed
view check	Passed
warning param check	Alarm View Details

Previous Verify Again **Start Task** Start Later

6. Return to the data migration task list, and you can see that the task has entered the **Creating** status. After 1–2 minutes, the data migration task will be started.

- Select **Structural migration** or **Full migration**: once completed, the task will be stopped automatically.
- Select **Full + Incremental migration**: after full migration is completed, the migration task will automatically enter the incremental data sync stage, which will not stop automatically. You need to click **Complete** to manually stop the incremental data sync.
 - Select an appropriate time to manually complete the incremental data sync and perform business switch.
 - You can see that the migration task is in the incremental sync stage, and there is no latency. Stop writing the source database for several minutes.
 - When the source-target database data gap is 0 MB, and the source-target database time lag is 0s, manually complete the incremental sync.



The screenshot shows a table of migration tasks in the Tencent Cloud console. The table has columns for Task ID / Name, Task Status / Progress, Running Mode, Last Check Result, Source Database Type, Target Database Type, Source Access Type, Address, Creation Time, and Operation. A single task is listed with ID 'cdt-j80gq' and name 'test-stacy'. Its status is 'Task successful' with a green progress bar. The running mode is 'Immediate execution' and the last check result is 'no job'. Both source and target database types are 'MySQL'. The source access type is 'Database' and the address is 'Source: cdb-reqmpwvx; Target: cdb-1mddsluj'. The creation time is '2021-12-01 19:29:53'. The operation column contains 'Immediate start', 'Pause', and 'More'.

Task ID / Name	Task Status / Progress	Running Mode	Last Check Result	Source Database Type	Target Database Type	Source Access Type	Address	Creation Time	Operation
cdt-j80gq test-stacy View Details	Task successful Current step: binlog_sinker Status: Task completed Start: 2021-12-01 19:28:46 End: --	Immediate execution	no job	MySQL	MySQL	Database	Source: cdb-reqmpwvx Target: cdb-1mddsluj	2021-12-01 19:29:53	Immediate start, Pause, More

7. (Optional) If you want to view, delete, or perform other operations on a task, click the task and select the target operation in the **Operation** column. For more information, see [Task Management](#).

Business cutover

After the migration task status becomes **Task successful**, you can formally cut over the business. For more information, see [Cutover Description](#).