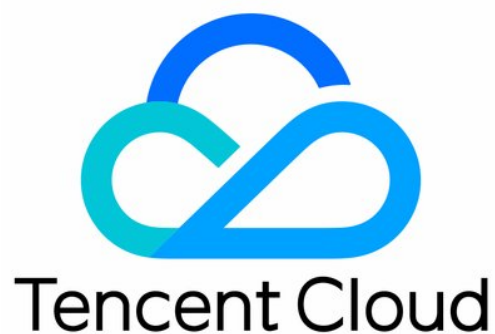


Data Transfer Service

Data Migration (OldDTS)

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

Data Migration (OldDTS)

MySQL Data Migration

- Migration between TencentDB for MySQL Instances

- Online Import of MySQL Data

MariaDB Data Migration

Redis Data Migration

- Migrating with DTS

- Migration with redis-port

Data Migration (OldDTS)

MySQL Data Migration

Migration between TencentDB for MySQL Instances

Last updated : 2023-06-14 10:33:54

This document describes how to migrate data between TencentDB for MySQL instances, using the data migration feature of Data Transmission Service (DTS). DTS supports structural migration, full data migration, and incremental data migration, as well as non-stop smooth data migration between TencentDB for MySQL instances.

Prerequisites

- You have [created a TencentDB for MySQL instance](#). Supported versions: MySQL 5.5, MySQL 5.6, and MySQL 5.7.
- You need to create a migration account on the target MySQL instance. Account permissions required: all the read/write permissions for objects to be migrated.

Notes:

A DTS data migration task between TencentDB for MySQL instances involves two steps: cold backup data export and incremental data sync. Cold backup data export and migrated data comparison have certain effect on the load of the source database, so we recommended that you perform database migration during off-peak hours or in the standby database.

- During the migration verification task, `lower_case_table_name` is specified to check whether the configuration of the source and target instances is consistent. If not, an error will be reported. Please avoid any restart issue due to `lower_case_table_name` in advance.
- For a full instance migration task, you need to restart the target instance after the full backup is imported.

Supported Migration Types

- Structural migration: DTS allows you to migrate the structure definition of migration objects to the target instance. Currently, DTS supports the structural migration of an entire instance and specified tables.

- Full migration: DTS allows you to migrate the full data of the source MySQL instance to the target TencentDB for MySQL instance.
- Incremental sync: DTS reads and parses the binlog information of the source MySQL database based on full data migration, and synchronizes the incremental data of the source MySQL database to the target MySQL database.

Precheck Items

1. To avoid conflict, check if any tables with the same name exist in the target TencentDB for MySQL database.
2. Check database versions. MySQL 5.5, MySQL 5.6, and MySQL 5.7 can be migrated to cloud.
3. Check and ensure that the target TencentDB for MySQL database has a larger capacity than the source TencentDB for MySQL database does.
4. Create a migration account on the source TencentDB for MySQL database. Ignore this step if you already have an authorized account for data migration.

```
GRANT ALL PRIVILEGES ON *.* TO "migration account"@ "%" IDENTIFIED BY "migration password";  
FLUSH PRIVILEGES;
```

5. Confirm source MySQL database variables.

Run `SHOW GLOBAL VARIABLES LIKE 'XXX';` to view the global MySQL variables so as to check whether the database in the current state is suitable for migration:

```
server_id > 1  
log_bin = ON;  
binlog_format = ROW/MIXED  
binlog_row_image = FULL  
innodb_stats_on_metadata = 0  
wait_timeout is recommended to be greater than or equal to 3600 seconds, and must be less than 7200 seconds  
Set the same duration for interactive_timeout and wait_timeout
```

If the source instance is a slave instance, confirm the following parameters in the source instance:

```
log_slave_updates = 1
```

6. Modify the variables.

- a. For a self-created MySQL database, you need to modify the configuration file `my.cnf` of the source MySQL database and restart:

```
log-bin=[custom binlog file name]
```

- b. Modify configuration dynamically:

```
set global server_id = 99;  
set global binlog_format=ROW;  
set global binlog_row_image=FULL;  
set global innodb_stats_on_metadata = 0;
```

Directions

1. Log in to the [DTS data migration console](#) and click **Create Migration Task** to enter the migration task creation page.
2. Select the region of the target instance and click **Buy Now**.
3. Complete task configuration, source database settings, and target database settings on the “Set source and target databases” page. After the connectivity test for the source and target databases is passed, click **Create**.

Note :

If the connectivity test fails, follow the prompts to troubleshoot, resolve the problem, and try again.

- Task Configuration
 - Task Name: specify a name for the task.
 - Scheduled Execution: specify the start time of the migration task.
 - Tag: you can use tags to manage resources by category from different dimensions.
- Source Database Settings
 - Source Type: select “MySQL”.
- Target Database Settings
 - Select the target database instance and enter its account and password.

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

Note :

- The `character_set_server` and `lower_case_table_names` configuration items are migrated only when the entire instance is migrated.
- If the character set configuration of migrated tables in the source instance is different from that of the target instance, the character set configuration of the source instance is retained.

- Migration Type: structural migration, full migration, and full + incremental migration supported.
- Migration Object: the entire instance and specified objects supported.
- Overwrite Target Database with Source Database Root Account: root accounts are used to verify the security of the TencentDB for MySQL database. Without the root account of the source database, you cannot conveniently use the TencentDB for MySQL database subsequently. Therefore, you need to specify whether to overwrite the target database root account with the source one during the full instance migration process. Select **Yes** if you want to use the root account of the source database or if the root account is not configured for the target database. Select **No** if you want to retain the root account of the target database.
- Read-only Target Database: if this configuration item is selected, during data migration, data from the source database will be read-only in the target database and cannot be changed until you click the corresponding button to complete the migration.
- Data Consistency Check: you can select "full data consistency check" or "no check".

5. Verify the migration task on the "Verify task" page. After the task is verified, click **Start**.

There are 3 statuses for task verification:

- Passed: the verification is fully successful.
- Warning: the verification fails. Database operation may be affected during or after data migration, but the migration task can still be executed.
- Failed: the verification fails and the migration task cannot be executed. In this case, please check and modify the migration task information according to the error and then verify the task again. For more information on the failure causes, please see "Verification Failure Description".

After the verification is passed, return to the data migration list and click **Start** to start data migration immediately. Please note that if you have configured scheduled execution, the migration task will begin queuing and be executed at the specified time; otherwise, it will be executed immediately.

After the migration is started, you can view the corresponding migration progress under the migration task. The subsequent steps required for migration and the current stage will be displayed if you hover over the exclamation mark after the current step.

Note :

Due to system design limitations, multiple migration tasks submitted or queued at the same time will be performed in sequence by queuing time.

7. During the creation of the migration task, the incremental sync option is selected by default. When data migration is completed, the target TencentDB for MySQL database will be set as the slave database for the source database. The incremental data of the source database during migration will be synchronized to the target TencentDB for MySQL database via master/slave sync. In this case, any changes made to the source database will be synchronized to the target TencentDB for MySQL database.

Note :

Do not write data into the target database instance before terminating the sync. Otherwise, data comparison may fail due to data inconsistency between the source and target databases, resulting in a migration failure.

8. (Optional) To cancel an in-progress migration task, click **Cancel**.

Note :

- Data that has been synchronized to the target database will not be cleared if you click **Cancel**.
- Restarting the task may cause verification or task failure. You may need to manually clear all databases or tables that may cause conflicts in the target database before you can start the migration task again.
- When migrating a single table, make sure that all tables relying on foreign keys are migrated.

9. Complete the migration.

Note :

If the migration is in “not completed” status, the migration task will continue, so will data sync.

When the migration progress reaches 100%, the data gap between the target and source databases is 0 MB, and the time delay between them is 0 second, it means data has been synchronized. You can click **Complete** on the right to complete the migration task.

Online Import of MySQL Data

Last updated : 2020-12-01 16:12:12

Operation Scenarios

DTS provides data migration and continuous data replication from self-created MySQL databases to TencentDB, allowing you to migrate hot data without interrupting your business. MySQL with public IP/port, Direct Connect-based MySQL in local IDC, and CVM-based self-created MySQL can be migrated. Currently, DTS is supported for MySQL 5.7.

Preparations

Precautions

- A DTS data migration task involves two steps: cold backup data export and incremental data sync. As cold backup data export and data comparison after migration have certain impact on the load of the source database, you are recommended to perform database migration during off-hours or in the replica database.
- Migration of specified table
If `lower_case_table_name` is specified for migration, a migration verification task will be performed to check whether the source and target databases have the same configuration; and if not, an error will be displayed, helping you prevent any restart issues caused by `lower_case_table_name`.
- Migration of entire database
 - Before migrating configurations, if the source database has some parameters requiring database restart for them to take effect (e.g., `lower_case_table_name`), and their values are different from those in the target database, you need to configure restart of the target database.
 - After the full backup is imported, you need to restart the target database.
- SUPER privilege for the source database is required in some scenarios.

SUPER privilege for source database

SUPER privilege for the source database is not required in most scenarios and required only in the following scenarios:

- You have selected "Full Detection" as "Data Consistency Detection".
- During binlog sync, if you create an event in the source database, and an account that is not used for DTS data migration is specified as the `DEFINER` for this event, an error will occur if the SUPER privilege is unavailable.

Migratable databases

- CVM-based self-created MySQL databases in the classic network or VPCs can be migrated to TencentDB.
- MySQL databases with public IP/port can be migrated to TencentDB.
- Direct Connect-based or VPN-based MySQL databases can be migrated to TencentDB.

Precheck items

1. To avoid conflict, check whether the target database has a table with the same name as a table in the source database.
2. Check the database version. Data migration is supported for MySQL 5.1, 5.5, 5.6, and 5.7. As TencentDB no longer supports MySQL 5.1, you are recommended to upgrade MySQL 5.1 to MySQL 5.5 first and then migrate data to TencentDB for MySQL 5.5. You can also use DTS to directly migrate from local MySQL 5.1 to TencentDB for MySQL 5.5.
3. Check whether the capacity of the target database is greater than that of the source database.
4. Create a migration account on the source MySQL database (this is not required if you already have an authorized account for data migration).

```
GRANT ALL PRIVILEGES ON *.* TO "migration account"@"%" IDENTIFIED BY "migration password";  
FLUSH PRIVILEGES;
```

5. Confirm the source database's MySQL variables.

Run `SHOW GLOBAL VARIABLES LIKE 'XXX';` to view the global MySQL variables so as to check whether the database in the current state is suitable for migration:

```
server_id > 1  
log_bin = ON;  
binlog_format = ROW/MIXED  
binlog_row_image = FULL  
innodb_stats_on_metadata = 0  
You are recommended to set `wait_timeout` to above or equal to 3,600 seconds and mandatorily below 7,200 seconds.  
Set `interactive_timeout` and `wait_timeout` to the same value.  
If the source database is replica, confirm the following parameters in it:  
log_slave_updates = 1
```

6. Modify the variables.

- a. For a self-created MySQL database, modify the source database's MySQL configuration file `my.cnf` and restart the database:

```
log_bin=[custom binlog filename]
```

- b. Modify the configuration dynamically:

```
set global server_id = 99;  
set global binlog_format=ROW;  
set global binlog_row_image=FULL;  
set global innodb_stats_on_metadata = 0;
```

Directions

1. Create a migration task

Log in to the [DTS Console](#), enter the data migration page, and click **Create Migration Task**.

2. Set the source and target databases

Configure the task, source database, and target database. After the network connectivity test is successful, click **Create**.

a. Task settings

- Task Name: name the task.
- Scheduled Execution: specify the start time of the migration task.
- Tag: categorize and manage resources by various metrics.

b. Source database settings

Source Database Type: MySQL with public IP, CVM-based self-created MySQL, Direct Connect-based MySQL, and VPN-based MySQL are supported.

Source Database Type	Description
MySQL with public IP	It refers to a MySQL database that can be accessed through a public IP. The following information is required: <ul style="list-style-type: none">• MySQL server address• MySQL port• MySQL account• MySQL password
CVM-based self-created MySQL	You can migrate a MySQL database deployed on a CVM instance in the classic network or VPC by specifying the CVM instance ID. The following information is required: <ul style="list-style-type: none">• Region: all CVM-based self-created MySQL databases can be migrated to TencentDB for MySQL over the private network• CVM instance ID• MySQL port

	<ul style="list-style-type: none"> MySQL account MySQL password
Direct Connect-based MySQL	<p>After connecting a self-created MySQL database in your local IDC to Tencent Cloud through Direct Connect, you can use DTS to migrate data to Tencent Cloud. The following information is required:</p> <ul style="list-style-type: none"> Direct Connect gateway: it is the Direct Connect gateway used by the database server to connect to Tencent Cloud. For more information, please see Direct Connect Gateway VPC: it is the VPC where the Direct Connect gateway resides MySQL server address: it is the address of your MySQL server in the IDC. The IP mapped by the Direct Connect gateway will be accessed during DTS data migration MySQL port MySQL account MySQL password
VPN-based MySQL	<p>After connecting a self-created MySQL database in your local IDC to Tencent Cloud through VPN Connections or a CVM-based self-created VPN service, you can use DTS to migrate data to Tencent Cloud. The following information is required:</p> <ul style="list-style-type: none"> Region: currently, only VPN services in the same region as that of the target TencentDB instance are supported VPN type: you can select VPN Connections or CVM-based self-created VPN VPN gateway: you need to provide the VPN gateway information only if VPN Connections is selected. For more information, please see VPN VPC: it is the VPC where the VPN Connections instance resides MySQL server address: it is the address of your MySQL server in the IDC. The IP mapped by the Direct Connect gateway will be accessed during DTS data migration MySQL port MySQL account MySQL password

c. Target database settings

Select the target database instance and enter its account and password.

3. Set migration options and select migration objects

Select the database to be migrated (you can choose to migrate the entire database or only certain tables), create a migration task, and check the task information.

⚠ Note :

- The `character_set_server` and `lower_case_table_names` configuration items will be migrated only if the entire database is migrated.
- If the character set configuration of the migrated tables in the source database is different from that of the target database, the former will be retained.

- **Migration Type:** structure migration, full migration, and full + incremental migration are supported.
- **Migration Object:** you can migrate the whole instance or specific objects.
- **Overwrite Target Database with Source Database Root Account:** the root account is used for TencentDB security verification. If the source database root account does not exist, issues may occur when you use TencentDB after migration. Therefore, to migrate the whole instance, you need to specify whether the target database root account should be overwritten by the source database root account. If you want to use the source database root account or the target database does not have any root account, choose **Yes**; if you want to keep the target database root account, choose **No**.
- **Read-only Target Database:** if the read-only option is enabled, data migrated to the target database will remain read-only during the migration process and cannot be modified until you complete the migration.
- **Data Consistency Detection:** you can perform a full detection or do not perform any detection.

4. Verify the migration task

After the migration task is created, you need to click **Next step: verify task** to verify the task information. You cannot start the migration task until all the check items are passed. Then, click **Start Task**.

There are 3 statuses for task verification:

- **Passed:** the verification is fully successful.
- **Warning:** the verification failed. Database operation may be affected during or after data migration, but the migration task can still be executed.
- **Failed:** the verification failed, and the migration task could not be executed. In this case, please check and modify the migration task information according to the error and then verify the task again. For the cause of failure, click **View Details** to view the "Verification Details".

5. Start migration

After the verification is passed, you can click **Immediate start** in the migration task list to start data migration. Please note that if you have configured scheduled execution, the migration task will begin queuing and be executed at the specified time; otherwise, it will be executed immediately.

After the migration is started, you can view the corresponding migration progress under the migration task. The subsequent steps required for migration and the current stage will be displayed if you hover over the exclamation mark after the current step.

Note :

Due to system design limitations, multiple migration tasks committed or queued at the same time will be performed in sequence by queuing time.

6. Perform incremental sync

When a migration task is created, the incremental sync option is selected by default. When data migration is completed, the target database will be set as the replica of the source database, and new data in the source database during migration will be synced to the target database through source/replica sync. In this way, any modification of the source database will be synced to the target database.

Note :

Before closing the sync, do not write data into the target database; otherwise, data comparison may fail due to inconsistency between the source and target databases, resulting in migration failure.

7. Cancel migration (optional)

Note :

1. Data that has been synced to the target database will not be cleared if you click **Cancel**.
2. Restarting the task may cause the verification or task to fail. You may have to manually clear all databases or tables that may cause conflicts in the target database before you can start the migration task again.
3. When migrating a single table, make sure that all tables depended on by its foreign keys are also migrated.

To cancel an ongoing migration task, click **Cancel**.

8. Complete migration

Note :

If the migration is in "not completed" status, the migration task will continue, so will data sync.

After the migration progress is 100%, the data lag between target and source databases is 0 MB, and the time lag between them is 0 seconds, you can click **Complete** on the right to complete the migration task.

MariaDB Data Migration

Last updated : 2022-09-02 15:04:25

Preparations

You need to have the permissions of the source database, including RELOAD, PROCESS, REPLICATION SLAVE, LOCK TABLES, REPLICATION CLIENT, SHOW DATABASE, EVENT, and SELECT.

If you want to migrate views in the source database, you also need to have the SHOW VIEW permission.

Directions

1. Create a migration task

- (1) Log in to the [DTS console](#) and click **Create Migration Task** on the data migration page.
- (2) On the **Create Migration Task** page, select the types and regions of the source and target databases and click **Buy Now**.

Note :

Select the region with caution as it cannot be changed once the migration task is created.

2. Set the source and target databases

Enter the information of the task, source database, and target database.

Task settings

Enter the name of the migration task. If you want the migration task to be executed at a later time, you can set scheduled execution.

1 Set source and target databases

2 Select type and database list

3 Verification task

Task Settings

Task Name *

Scheduled Execution * ☒

Execution Time * 2019-11-09 00:00:00

Source Database Settings

Source Database Type * MySQL MariaDB PostgreSQL Percona

Access Type * Direct connect VPN access Cloud database

Region * South China (Guangzhou)

Account * 00:00:00

Password *

Test Connectivity

Nov 2019

Su	Mo	Tu	We	Th	Fr	Sa
					1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30

00:00:00

OK Cancel

Source database settings

Enter the source database information and click **Test Connectivity** to check whether the source database can be connected.

Source Database Settings

Source Database Type * MySQL Redis MongoDB MariaDB PostgreSQL Percona

Access Type * Public network CVM-based Direct connect VPN access Cloud database

Region * South China (Guangzhou)

Instance ID * tdsql-kijd6cth

Account * kaylalzhang

Password * *****

Test Connectivity

Target Database Settings

Target Database Type * MariaDB

Connection testing completed

Test Items	Test Results	Description
Telnet	Approve	
Mysql Connect	Approve	

Target database settings

Enter the target database information and click **Save**.

Target Database Settings

Target Database Type *

MariaDB

Region *

South China (Guangzhou) ▼

Instance ID *

Please select data ▼

Cancel

Save

3. Select the type and databases/tables

Select the type and database list and click **Next step: verify task**.

✓ Set source and target databases

2 Select type and database list

3 Verification task

Select a type *

Structure Transfer

Full Migration

All + Incremental Migration

Select an object *

Whole instance

Specify instance

Note

1. The recommended time for incremental migration is no more than 15 days. Please ensure that you click the migration completion button after master-slave catch-up, [Details](#)

Cancel

Back

Next step: verify task

4. Verify the task

Check whether the source instance runs normally and whether the sets to be migrated to the target instance are in conflicts.

✓ Set source and target databases

✓ Select type and database list

3 Verification task

Progress 100%

✓ Check parameter	Approve
✓ Query target instance info	Approve
✓ Apply for network access permission	Approve
✓ Try to connect to source instance	Approve
✓ Check source instance permissions	Approve
✓ Check source instance configuration	Approve
✓ Check compatibility	Approve
✓ Get the size of data that the source instance ...	Approve
✓ Check the remaining space of the target mac...	Approve
❗ Check if target instance is empty	Failed View Details
✓ Check if the target instance space is sufficient	Approve
✓ Check target instance log filtering	Approve
✓ Check the foreign key engine, etc.	Approve
✓ Release network access permission	Approve

Cancel

Verify Again

Back

Start

5. Complete the migration

After the verification is passed, return to the migration task list. After the incremental sync is 90% complete, click **Done** on the right of the migration task.

ID/Name	Status	Type	Address	Creation Time	Operation
dts-g- -	Preparation completed	Source: mariadb Target: mariadb	Source: tdsq-l- Target: tdsq-l-	2019-11-11 13:05:31	View Modify Verification Delete Start Now Cancel Complete Create Similar Task
2019-11-11 13:11:26 Start					
Build master-slave log sync 99% Build master-slave log sync (5/6 steps) Progress: 99%					
Not finished					

Complete the migration.

ID/Name	Status	Type	Address	Creation Time	Operation
dts- <div></div>	Preparation completed	Source: mariadb Target: mariadb	Source: tdsq- <div></div> Target: tdsq- <div></div>	2019-11-11 13:05:31	View Modify Verification Delete Start Now Cancel Complete Create Similar Task
2019-11-11 13:11:26 Start					
<div><div>Build master-slave log sync 99%</div><div>Build master-slave log sync (5/6 steps ⓘ)</div><div>Progress: 99%</div></div>					
Not finished					

Redis Data Migration

Migrating with DTS

Last updated : 2022-09-21 18:04:04

Background Overview

DTS is a data transmission service that integrates such features as data migration, sync, and subscription. It helps you migrate your databases to the cloud without interrupting your business and build a high-availability database disaster recovery architecture through real-time sync channels. Its data subscription feature meets your requirements for commercial data mining and async business decoupling.

DTS for Redis currently supports the data migration feature for you to migrate data to TencentDB in a non-stop manner at a time. In addition, in its full + incremental data migration mode, historical data in the source database written before migration and incremental data written during migration can be migrated together.

Use Cases

DTS for Redis data migration supports source and target databases in the following deployment modes:

Source Database	Target Database	Description
Self-built Redis database, such as self-built databases in IDC and CVM	TencentDB for Redis	-
Third-party Redis	TencentDB for Redis	The third-party cloud vendor must grant the <code>SYNC</code> or <code>PSYNC</code> command permission.
TencentDB for Redis	TencentDB for Redis	Data migration, version upgrade, and cross-region migration are supported between database instances under the same Tencent Cloud account.

Migration Support Description

Note :

For compatibility issues with migration from Standalone Edition to Memory Edition (Cluster Architecture), see [Check on Migration from Standard Architecture to Cluster Architecture](#).

Supported versions

- DTS supports Redis 2.8, 3.0, 3.2, 4.0, 5.0, and 6.0. To use it for v6.0, [submit a ticket](#) for application. We recommend you migrate from an earlier version to a later version; otherwise, compatibility problems may occur.
- Standard architecture and cluster architecture instances can be migrated to each other; however, such heterogeneous migration may have compatibility problems.
- Supported architectures include single-node, Redis cluster, Codis, and twemproxy.
- Migration permission requirements: To migrate data through DTS, the source instance must support `SYNC` or `PSYNC` commands.

Supported networks

DTS supports data migration based on the public network, CVM instances, Direct Connect gateways, VPN, and CCN.

- Public Network: The source database can be accessed through a public IP.
- Self-Build on CVM: The source database is deployed in a [CVM](#) instance.
- Direct Connect: The source database can be interconnected with VPCs through [Direct Connect](#).
- VPN Access: The source database can be interconnected with VPCs through [VPN Connections](#).
- CCN: The source database can be interconnected with VPCs through [CCN](#).

Limitations

- To ensure migration efficiency, cross-region migration of CVM-based self-built instances is not supported.
- To migrate instances over the public network, make sure that the source instance can be accessed over the public network.
- Only instances that are running normally can be migrated, while instances with no password initialized or with ongoing tasks cannot.
- The target instance must be empty. During the migration process, the target instance will be read-only and cannot be written to.
- After the successfully migrated data is verified by your business, you can disconnect the source instance and connect to the target instance.

Environment Requirements

System check

Note :

The DTS system will check the following environment requirements before starting a migration task and report an error if a requirement is not met. You can also check them in advance. For more information on error handling, see [Check Item Overview](#).

Type	Environment Requirement
Requirements for source database	<ul style="list-style-type: none">• The source and target databases can be connected.• The number of databases in the source database must be less than or equal to that in the target database.• The source instance must be on v2.2.6 or later.• The source database must be a replica node.
Requirements for target database	<ul style="list-style-type: none">• The target database version must be later than or equal to the source instance version; otherwise, an alarm will be triggered for compatibility problems during verification.• The target database must have the latest proxy.• The space of the target database must be at least 1.5 times the volume of the data to be migrated in the source database.• The target database must be empty.

Manual check

You should check and make sure that the following items are passed before the migration; otherwise, the migration may fail.

Big keys in the source database

Before the migration, check whether there are big keys in the source database. They may cause the buffer (client-output-buffer-limit) to overflow during the migration, leading to a migration failure.

- For TencentDB databases, you can use the performance optimization feature of TencentDB for DBbrain to quickly analyze big keys. For more information, see [Memory Analysis](#).
- For non-TencentDB databases, use RDBTools to analyze big keys in Redis.

Evaluate large keys for splitting or cleaning. If you need to retain them, set the source buffer size (client-output-buffer-limit) to infinite.

```
config set client-output-buffer-limit 'slave 0 0 0'
```

Limit on the number of TCP connections in the source Linux kernel

If the number of concurrent business requests is high, check the limit on the number of connections in the Linux kernel before the migration. If this value is exceeded, the Linux server will actively disconnect from DTS.

```
echo "net.ipv4.tcp_max_syn_backlog=4096" >> /etc/sysctl.conf
echo "net.core.somaxconn=4096" >> /etc/sysctl.conf
echo "net.ipv4.tcp_abort_on_overflow=0" /etc/sysctl.conf
sysctl -p
```

Access permission of the source RDB file directory

Before the migration, check and make sure that the directory where RDB files are stored in the source database is readable; otherwise, the migration will fail.

If the RDB file directory is not readable, run the following command in the source database to set "diskless replication". Then, RDB files will be directly sent to DTS for storage, with no need to be stored in the source database first and then sent.

```
config set repl-diskless-sync yes
```

Command compatibility (for migration from standard architecture to cluster architecture)

The most challenging problem in migrating data from Standard Edition to Memory Edition (Cluster Architecture) is the command compatibility with usage specifications of Memory Edition (Cluster Architecture).

- Multi-key operations

TencentDB for Redis Memory Edition (Cluster Architecture) only supports cross-slot multi-key access for `mget`, `mset`, `del`, and `exists` commands. In the source database, keys that need to engage in multi-key computing can be aggregated into the same slot through a hash tag. For more information on how to use hash tags, see [Redis cluster specification](#).

- Transactional operations

Memory Edition (Cluster Architecture) supports transactions, but cross-slot access to keys in transactions is not supported.

Perform static and dynamic evaluations as instructed in [Check on Migration from Standard Architecture to Cluster Architecture](#).

Migration Directions

1. Create a migration task

- (1) Log in to the [DTS console](#) and click **Create Migration Task** on the data migration page.
- (2) On the **Create Migration Task** page, select the types and regions of the source and target databases and click **Buy Now**.

2. Set source and target databases

Configure the **Source Database Settings** and **Target Database Settings** and click **Test Connectivity**. After the test is passed, click **Save** to proceed to the next step.

1 Set source and target databases > **2 Verification task**

Task Settings

Task Name *

Scheduled Execution * ☐

Source Database Settings

Source Database Type * MySQL **Redis** MongoDB MariaDB PostgreSQL Percona

Access Type * **Public network** CVM-based Direct connect VPN access Cloud database Cloud connect network

Node Type * Single-node Migration **Cluster Migration**

Region * Select a region

Node Info * Please enter the node info: IP, port, password. One node per line (use commas and quotation marks). To ensure the stability of source service, please migrate data from source Redis slave database. For example, 186.3.55.77,4100,"zxczs_6688"

Setting Type	Configuration Item	Description
Task Configuration	Task Name	Set a task name that is easy to identify.
	Running Mode	<p>You can set Immediate execution or Scheduled execution.</p> <ul style="list-style-type: none"> If a scheduled task is modified and passes verification, you need to click Scheduled start again to make the task start at the scheduled time. If the specified time has passed, the task will be started immediately. You can also click Immediate start to start the task immediately.

Setting Type	Configuration Item	Description
	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	The source database type selected during purchase, which cannot be changed.
	Region	The region selected during purchase, which cannot be changed.
	Access Type	<p>For a third-party cloud database, you can select Public Network generally or select VPN Access, Direct Connect, or CCN based on your actual network conditions.</p> <p>In this scenario, select Direct Connect or VPN Access. You need to configure VPN-IDC interconnection in this scenario. For the preparations for different access types, see Overview.</p> <ul style="list-style-type: none"> Public Network: The source database can be accessed through a public IP. Self-Build on CVM: The source database is deployed in a CVM instance. Direct Connect: The source database can be interconnected with VPCs through Direct Connect. VPN Access: The source database can be interconnected with VPCs through VPN Connections. Database: The source database is a TencentDB instance. CCN: The source database can be interconnected with VPCs through CCN.
	VPC-based Direct Connect Gateway	Only VPC-based Direct Connect gateway is supported. Confirm the network type associated with the gateway.
	VPC	Select a VPC and subnet associated with the VPC-based Direct Connect gateway.
	Node Type	<p>Single-Node Migration and Cluster Migration are supported. Cluster Migration is used as an example here.</p> <p>Currently, there are no limits on the number of shards and replicas in migration from Cluster Edition Redis to Cluster Edition Redis.</p>

Setting Type	Configuration Item	Description
	Node Info	Enter the addresses and passwords (IP:port:password or IP:port) of all shards of the source database cluster and separate the information of different nodes with line breaks. We strongly recommend you migrate data from a replica node of the source database to avoid any impact on business access to the source database.
Target Database Settings	Target Database Type	The target database type selected during purchase, which cannot be changed.
	Region	The target database region selected during purchase, which cannot be changed.
	Access Type	Select a type based on your scenario. In this document, select Database .
	Database Instance	Select the instance ID of the target database.

3. Verify and start the task

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

- Failed: It indicates that a check item fails 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.

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

4. Complete the migration task

(1) (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 [Viewing Task](#).

(2) If the keys of the source and target databases are the same, click **Done** in the **Operation** column to stop the data migration task.

(3) After the migration task status becomes **Task successful**, verify the data in the target database. If the verification is passed, you can formally cut over the business. For more information, see [Cutover Description](#).

Event Alarming and Metric Monitoring

(1) DTS can automatically report event alarms triggered upon migration interruption to keep you informed of any exceptions. For detailed directions, see [Configuring Alarm Policy for Data Migration](#).

(2) DTS allows you to view the monitoring data of various metrics during migration to understand the performance metrics of the system. For more information, see [Viewing Monitoring Metric](#).

Migration with redis-port

Last updated : 2021-02-02 09:48:51

Overview

Download [redis-port \(Linux 64-bit\)](#).

redis-port is a collection of open-source tools mainly used for database sync, data import, and data export between Redis nodes and supports cross-version Redis data migration. The toolkit contains the following tools:

- redis-sync: it is used for data migration between Redis instances.
- redis-restore: it supports importing Redis backup files (in RDB format) to specified Redis instance.
- redis-dump: it supports backing up Redis data in RDB format.
- redis-decode: it supports decoding Redis RDB backup files into readable files.

Compatible Versions

- Source instances on Redis 2.8, 3.0, 3.2, and 4.0 are supported.
- Target instances on Redis 2.8, 3.0, 3.2, and 4.0 and in all editions of TencentDB are supported, including Redis Memory Edition and CKV Edition.

Online Migration Through redis-sync

How it works

- redis-sync has two modules which are simulated as replication nodes to sync data from the source instance and translate the replicated data into write commands to update the target instance.
- Data replication is done in two phases: full sync and incremental sync.

Parameter description:

- -n: number of concurrent write tasks. You are recommended to leave it empty or set it to CPU core quantity * 2.
- -m: source instance address in the format of "password"@ip:port or ip:port (in password-free mode).
- -t: target instance address in the format of "password"@ip:port or ip:port (in password-free mode).
- --tmpfile=FILE: temporary filename.
- --tmpfile-size=SIZE: maximum size of the temporary file.
- --help: help command.

Sample:

```
./redis-sync -m 127.0.0.1:6379 -t "xxx2018"@10.0.5.8:6379
```

Output log:

```
[root@VM_5_16_centos bin]# ./redis-sync -m 127.0.0.1:6379 -t "xxx2018"@10.0.5.8:6379
2019/02/21 09:56:00 sync.go:76: [INFO] sync: master = "127.0.0.1:6379", target = "xxx2018@10.0.5.8:6379"
2019/02/21 09:56:01 sync.go:103: [INFO] +
2019/02/21 09:56:01 sync.go:109: [INFO] sync: runid = "f63e2ad58e2fcc15c8cc122f15778389a012c1a4", offset = 18576271
2019/02/21 09:56:01 sync.go:110: [INFO] sync: rdb file = 9063349 (8.64mb)
2019/02/21 09:56:01 sync.go:208: [INFO] sync: (r/f,s/f,s) = (read,rdb.forward,rdb.skip/rdb.forward,rdb.skip)
2019/02/21 09:56:02 sync.go:250: [INFO] sync: rdb = 9063349 - [100.00%] (r/f,s/f,s)=(1703936/71754,0/0,0) ~ (1.62mb/-,-/-,-) ~ speed=(1.62mb/71754,0/0,0)
2019/02/21 09:56:03 sync.go:250: [INFO] sync: rdb = 9063349 - [100.00%] (r/f,s/f,s)=(3407872/153850,0/0,0) ~ (3.25mb/-,-/-,-) ~ speed=(1.62mb/82096,0/0,0)
2019/02/21 09:57:54 sync.go:250: [INFO] sync: rdb = 9063349 - [100.00%] (r/f,s/f,s)=(80487526/411969,0/1587212,0) ~ (76.76mb/-,-/-,-) ~ speed=(0/0,0/0,0)
```

Use instructions:

- The database capacity of the target instance should be greater than that of the source instance; otherwise, the migration will fail.
- If migration is interrupted for reasons such as network failure, you need to empty the target instance first and then perform migration again; otherwise, there may be dirty data.
- The progress of migration is displayed in the log, where "sync: rdb = 9063349 - [100.00%]" indicates that full data has been synced and incremental data sync is in progress, while "speed=(0/0,0/0,0)" indicates that incremental data has been synced.
- You can stop data sync and migration by pressing Ctrl + C or through other means.

Importing Data Through redis-restore

redis-restore supports importing Redis backup files (in RDB format) on Redis 2.8, 3.0, 3.2, and 4.0 as well as AOF files into the specified Redis instance.

Parameter description:

- -n: number of concurrent write tasks. You are recommended to leave it empty or set it to CPU core quantity * 2.
- -i: RDB file path.

- -t: target instance address in the format of "password"@ip:port or ip:port (in password-free mode).
- -a: AOF file path.
- --db=DB: database ID of the target Redis instance for backup file import, which should be the same as that of the source instance.
- --unixtime-in-milliseconds=EXPR: the Key expiration time value is updated in the process of data import.
- --help: help command.

Sample:

```
./redis-restore dump.rdb -t 127.0.0.1:6379
```

Backing up Data Through redis-dump

redis-dump supports backing up Redis data into RDB files and incremental data into AOF files.

Note :

TencentDB for Redis currently does not support backing up data through redis-dump. You can back up and download data in the TencentDB for Redis console or through APIs. However, you can use redis-dump to back up your self-built Redis instances.

Parameter description:

- -n: number of concurrent write tasks. You are recommended to leave it empty or set it to CPU core quantity * 2.
- -m: Redis instance address in the format of "password"@ip:port or ip:port (in password-free mode).
- -o: path of the output RDB file.
- -a: Path to the outputted AOF file.
- --help: Help command.

Sample:

```
./redis-dump 127.0.0.1:6379 -o dump.rdb
```