TencentDB for MongoDB

FAQs

Product Documentation

Contents

FAQs

Cost

Last updated：2024-01-15 14:49:56

How do I purchase a TencentDB for MongoDB instance?

You can purchase an instance directly on the TencentDB for MongoDB purchase page as instructed in Creating TencentDB for MongoDB Instance.

Features

Last updated：2024-01-15 14:49:56

How do I get the slow logs of an instance?

You can use the slow log query feature as described in Slow Log Management to get the slow log details.

Does MongoDB support password-free access?

TencentDB for MongoDB supports auth-free access to instances. This feature enables you to efficiently and quickly access database instances but brings security risks. Therefore, enable it with caution.

How do I set secondary database dump?

In mongodump parameters, set `readPreference=secondaryPreferred`.

Does TencentDB for MongoDB support adding secondary nodes dynamically?

No.

What is the difference between TencentDB for MongoDB and self-built MongoDB?

For more information, see Strengths.

What is the oplog size? Can I adjust it?

The default oplog size is 10% of the instance capacity. You can adjust it between 10% (minimum) and 90% (maximum) of the instance capacity in the console.

Is oplog included in the purchased capacity?

As oplog exists inside the MongoDB database, it will occupy a part of the purchased capacity (10% by default).

What role permissions are available currently?

Only RoleDBAdminAny and RoleReadWriteAny are available, while root privileges are unavailable for now. More roles will become available in the future. We will also open up more convenient and practical console features to eliminate the need to call for some special permissions.

What will happen if disk utilization reaches 100%?

At this point, the instance will be blocked and become read-only, and any connections for write attempts will be closed. Pay attention to your business development and instance usage and expand the capacity as appropriate when capacity usage reaches a certain level.

Why does the monitor show high memory utilization in MongoDB?

MongoDB uses a greedy policy to try to allocate available memory for caching to improve performance. For more information, see FAQ: MongoDB Storage.

Which engines are supported by MongoDB?

WiredTiger engine and Rocks engine are supported currently.

Does MongoDB support maintenance window?

You can adjust the instance maintenance time in the TencentDB for MongoDB console as instructed in Setting Instance Maintenance Time.

Why is the space not reclaimed after data in MongoDB is deleted?

Unless a database or collection is dropped directly, the space freed by deleted data will not be reclaimed in MongoDB. To reclaim the space of the WiredTiger engine, see FAQ: MongoDB Storage.

Sharded Cluster

Last updated：2024-01-15 14:49:55

Does TencentDB for MongoDB support sharding?

Yes. For more information, see Creating TencentDB for MongoDB Instance.

What is a MongoDB sharded cluster?

TencentDB for MongoDB provides sharded clusters.

These sharded clusters distributively store data in multiple physical machines based on shardkey. Their great scalability, therefore, makes them very suitable for storing terabytes to petabytes of data.

Sharded clusters support instance-level backup and Rollback to ensure high data reliability. Multi-node automatic disaster recovery mechanism is used in each shard to ensure high service availability.

You can leverage the sharding feature of TencentDB for MongoDB to build a massive distributed storage system easily and efficiently.

How do I create a MongoDB sharded cluster?

Log in to the TencentDB for MongoDB purchase page, click **Sharded Cluster** in **Instance Type**, and select the number of shards, the number of nodes in each shard, and node specification, according to your needs.

Each shard is a replica set containing multiple nodes. Multi-node automatic disaster recovery mechanism is used in each shard to ensure high service availability.

How do I query the information of a MongoDB sharded cluster?

In the console, you can view detailed information of the sharded cluster instance, such as the shard composition, shard node specifications, and used capacity, as well as perform operations including instance capacity expansion.

How do I expand the capacity of a MongoDB sharded cluster?

You can expand the capacity only by expanding all nodes at the same time. Adding nodes is not supported.

In the console, click **Adjust Configuration** on the instance list page and select the desired capacity.

How do I monitor data in a MongoDB sharded cluster instance?

You can monitor the data of the TencentDB for MongoDB sharded cluster in three dimensions:

Instance

Shard

Node

You can view the monitoring report on the **System Monitoring** page of an instance, from which you will see various metrics, including operation requests, capacity usage, and load.

What is the MongoDB sharding policy?

Hash key sharding mechanism is supported.

The shardkey is indexed compound fields.

Sharding is required for all data sets in a shard instance. We recommend that you store non-sharded data in a separate replica set instance.

What is the authentication mechanism for MongoDB shards?

MongoDB is fully compatible with SCRAM-SHA-1 and MONGODB-CR.

What sharded cluster commands are supported by MongoDB?

For more information, see Command Support in Sharded Cluster v3.2.

How do I select the sharding method for a sharded cluster instance?

You need to enable sharding for data collections in advance in the sharded cluster instance. The shardkey plays a crucial role in the read/write performance of the cluster. On TencentDB for MongoDB earlier versions, the shardkey cannot be modified once set. Generally, you can select an optimal shardkey as follows:

Make sure that the written data can be evenly distributed across multiple shards.

Make sure that frequent queries can get data from the same shard as much as possible, so as to avoid aggregation in mongos after data is obtained from multiple shards.

For frequent range queries, we recommend that you use ranged sharding.

For frequent queries with specified fields, we recommend that you use hashed sharding.

How do I perform presharding in a sharded cluster instance?

For a sharded cluster instance, presharding can ensure that data will be written into multiple shards as evenly as possible while avoiding business jitters caused by the `moveChunk` operation due to the imbalance between different shard chunks during peak hours. TencentDB for MongoDB supports two sharding strategies for data presharding:

Hashed presharding

The hashed presharding feature is supported by default, and you can run the following command to implement it (here, `n` is the number of shards):
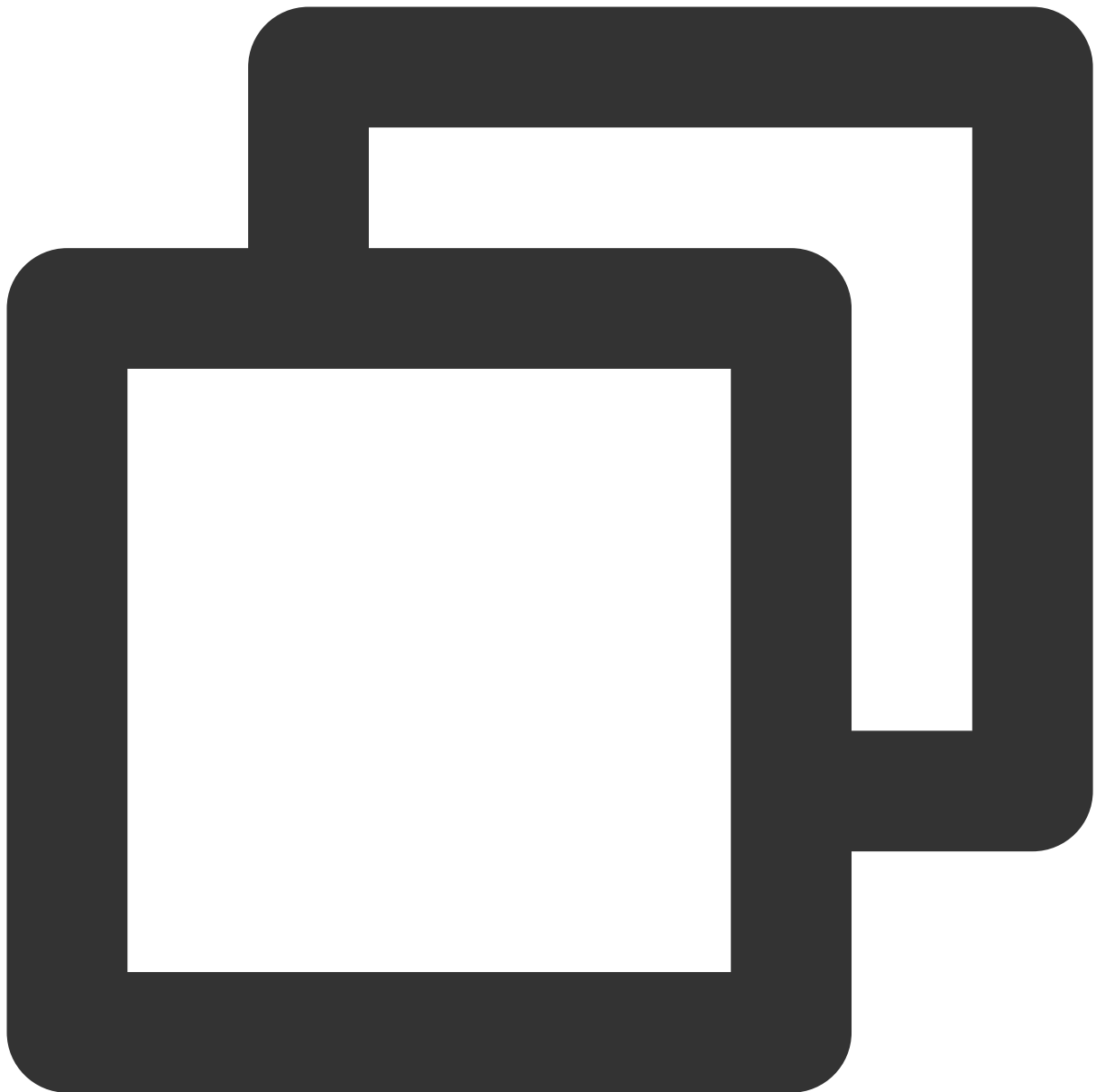
```
sh.shardCollection("db.collection", {bookId:"hashed"}, false, { numInitialChunks: 8
```

Ranged presharding

MongoDB natively does not support ranged presharding by default, because it cannot know your data distribution.

You can use **splitAt** for presharding to split the data into two parts within the specified ranges. For more information, see Split Chunks in a Sharded Cluster.

Balance window settings

If chunks are unevenly distributed across shards, `moveChunk` will be triggered to balance data. This process

involves data migration, which increases the system load and affects the business during peak hours. Therefore, we recommend that you perform data balancing during off-peak hours and set the balance window. Below are examples:
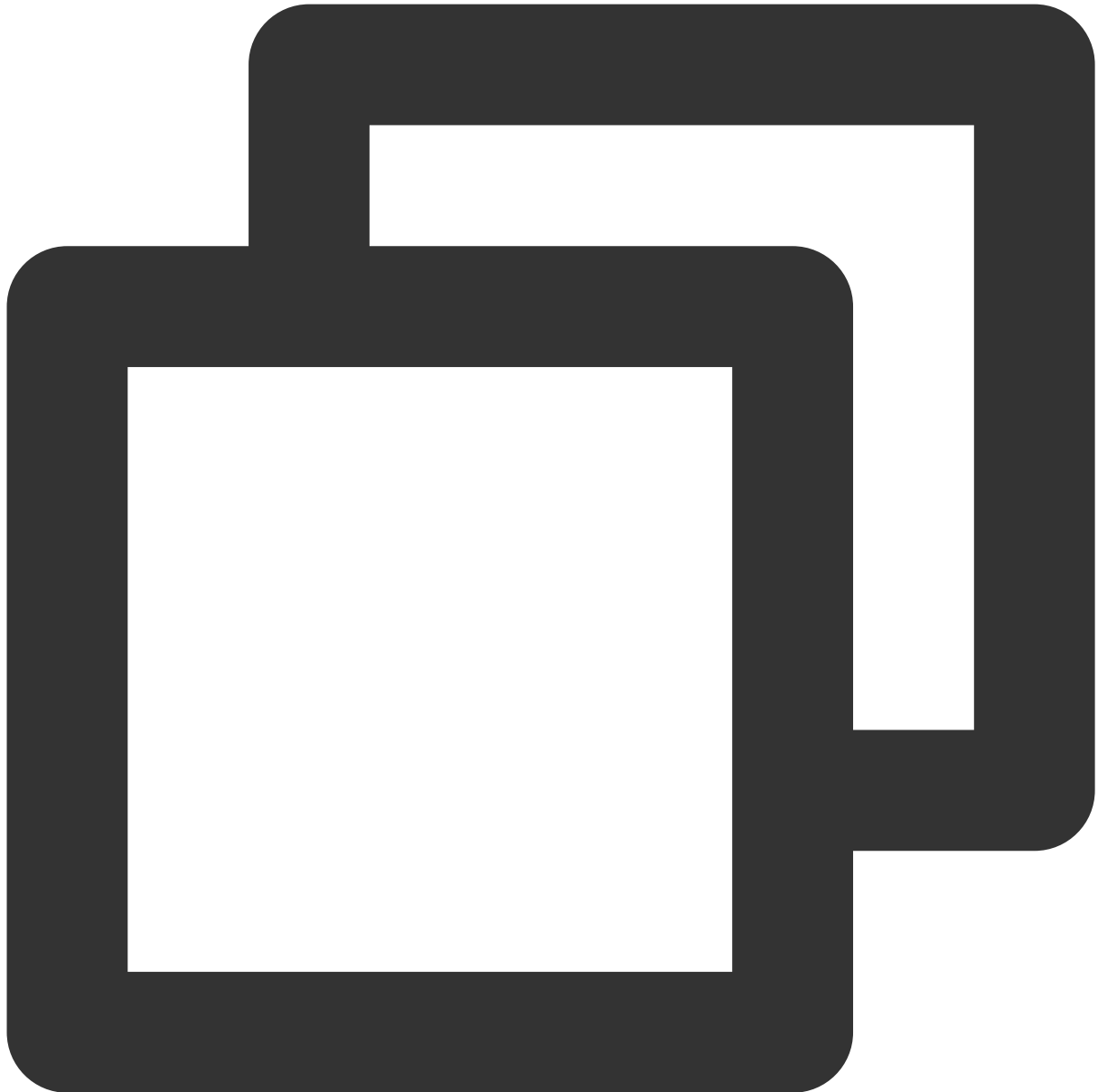

```
use config
db.settings.update({"_id":"balancer"},{"$set":{"activeWindow":{"start":"02:00","sto
```

How do I solve the problem where the database performance is compromised when a query request without the shardkey is executed in a sharded cluster instance?

Issue

If a query request carries the shardkey, it can be ensured that the data in the corresponding shard will be read, which maximizes the read performance. However, in actual business scenarios, when a business accesses the same table, some query requests carry the shardkey field while others don't. Such other requests will be broadcast to multiple shards, and the queried data will be aggregated in mongos and then returned to the client, leading to a low query efficiency.

Solution

If the cluster has a high number of shards, and the frequency of query requests without the shardkey is also high, you can create a secondary index table to solve this problem.

For example, in a feed information flow business, each feed information entry corresponds to a detail data entry and is stored in the `feed_info` details collection as shown below:
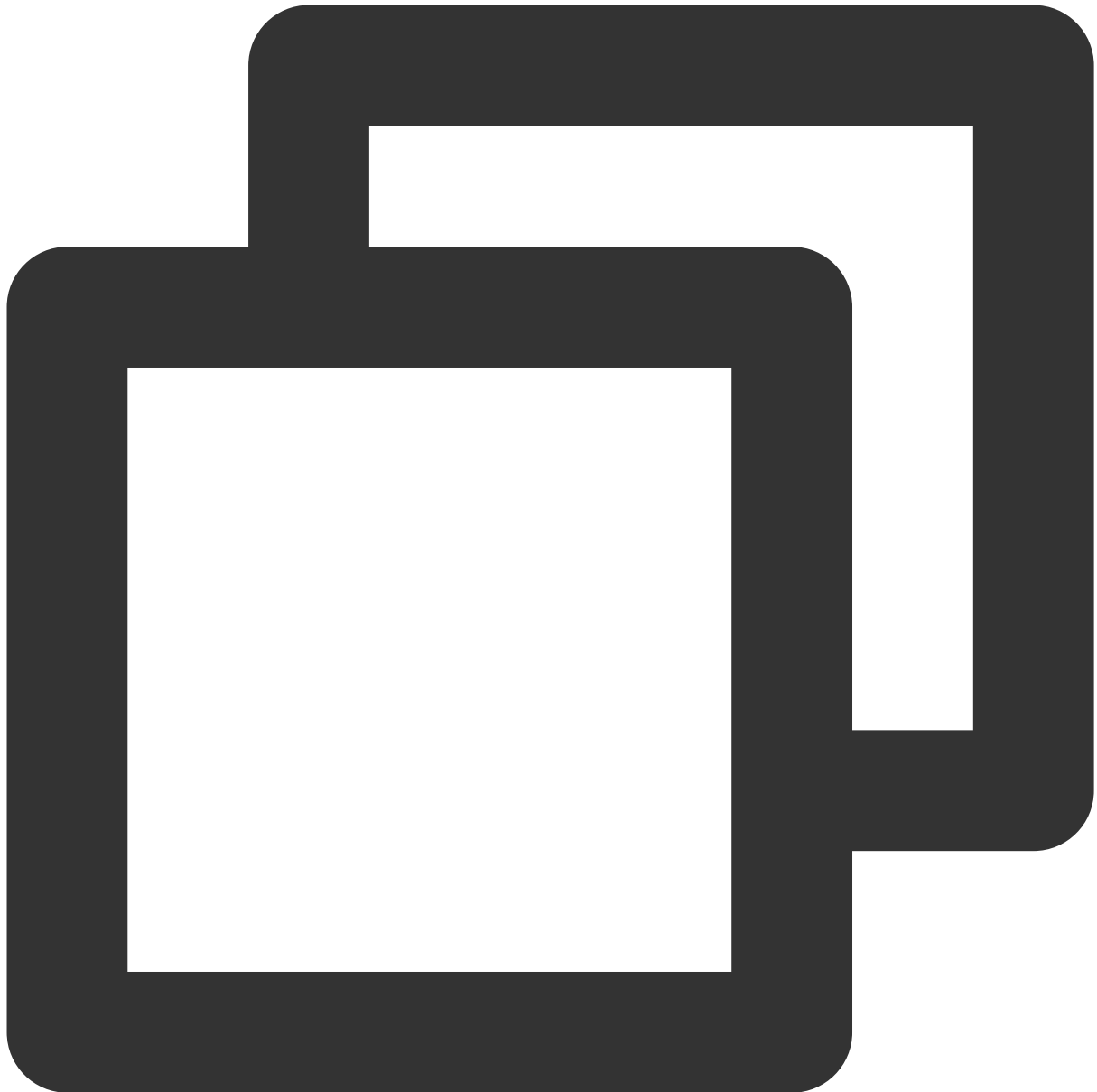
```
{
"_id" : ObjectId("614498889f97fd4692d22991")
"feedId": "223"
"userId": 3445
"feedInfo": "xxxxxxx"
}
```

As the `feed_info` table frequently queries data by `userId`, `userId` is used as the shardkey. However, the business also uses `feedId` frequently for query besides `userId`. As queries by `feedId` don't carry the shardkey, they will be broadcast to all shards, and the performance will be low if there are many shards.

In this case, you can use the `FeedId_userId_relationship` secondary index collection with `feedId` being the shardkey. The mappings between this collection and the `feed_info` collection are as shown below:

Table mapping diagram
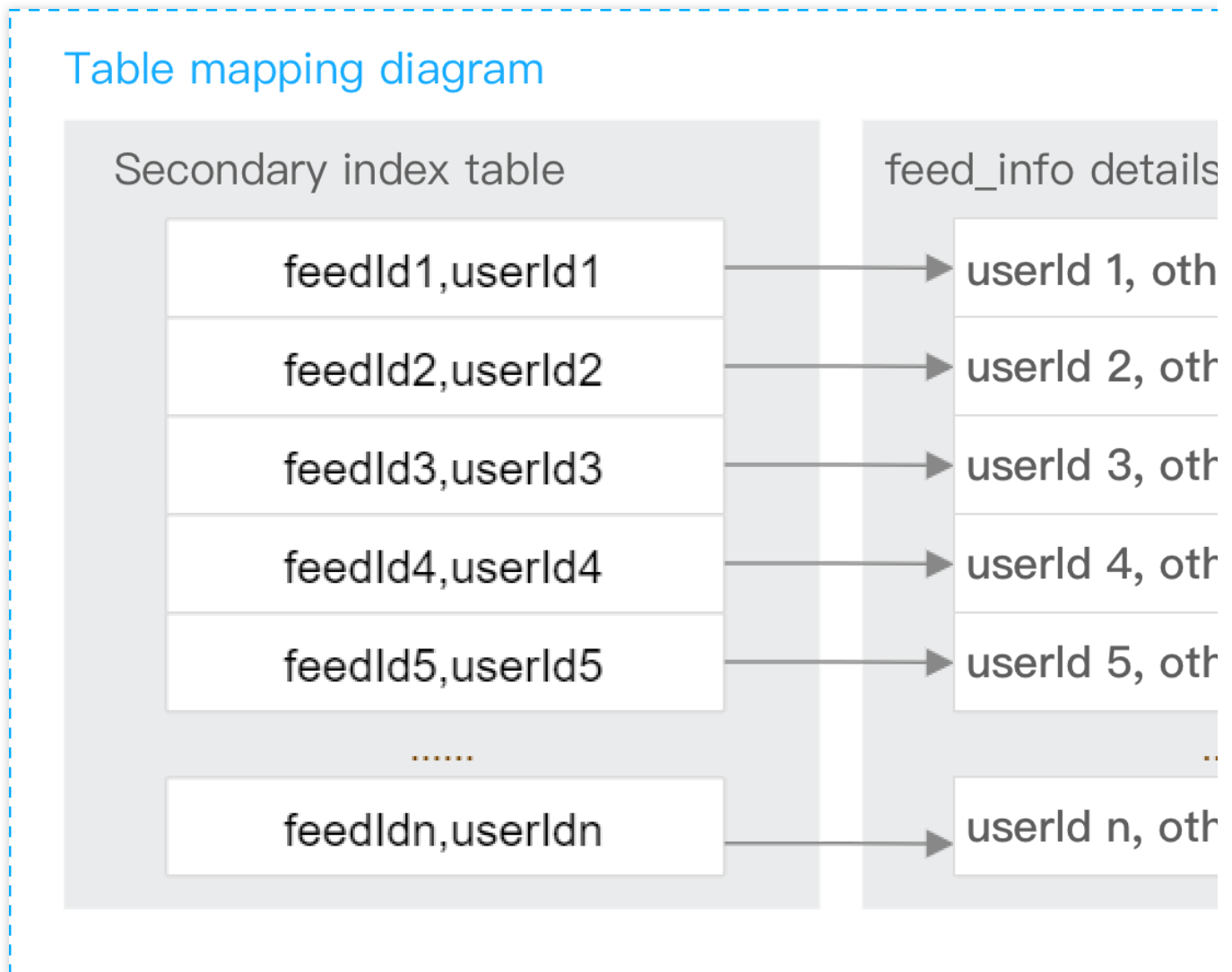
As shown above, if you use a `feedId` to query the information of a specific feed, the corresponding `userId` of the `feedId` will be found from the secondary index collection, and the found `userId` and `feedId` will be used together to get the required details. The entire query process involves two collections, and the query statements are as follows:
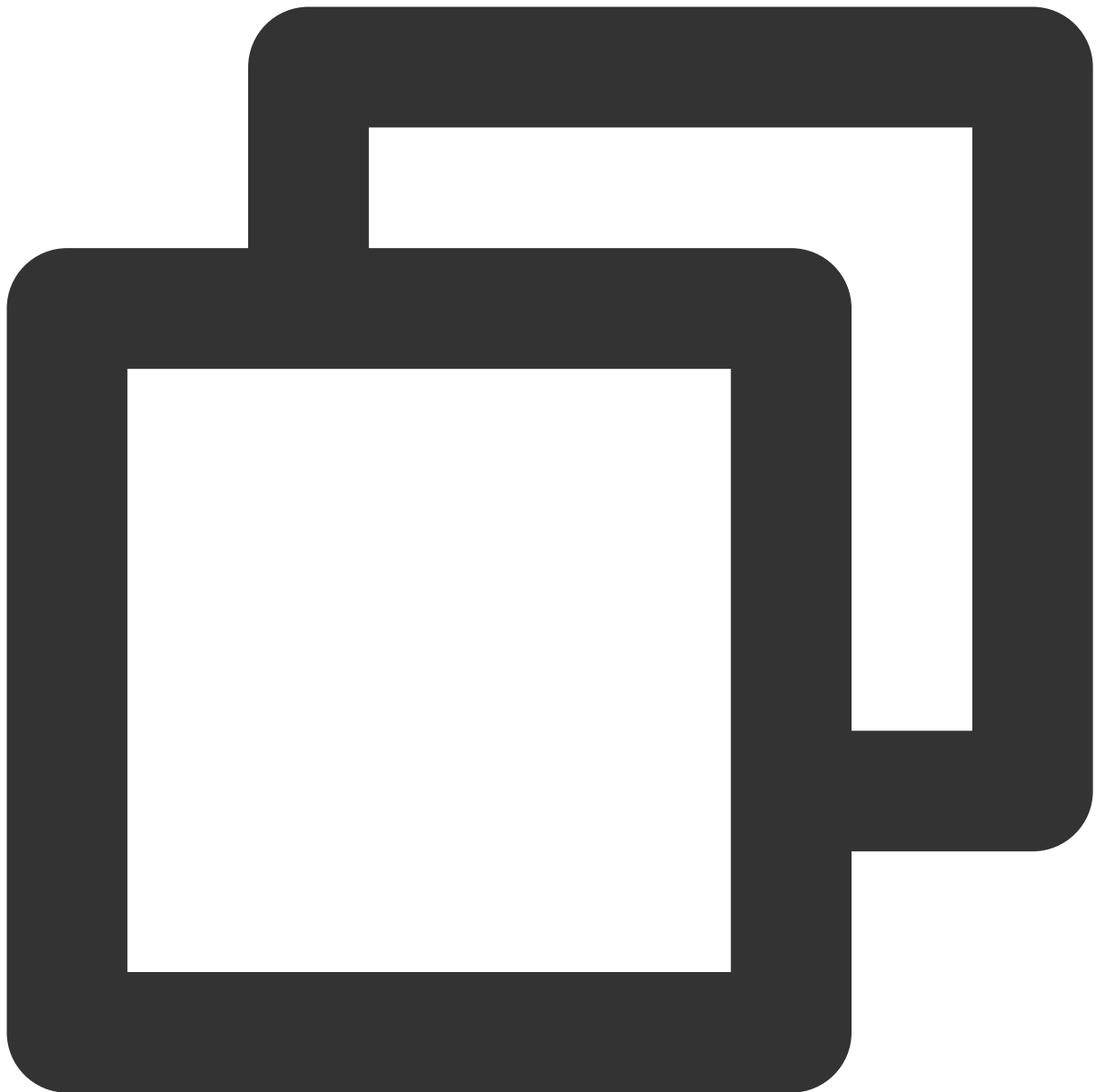
```
1.    // The shardkey of the `FeedId_userId_relationship` collection is `feedId`,
2.    db.FeedId_userId_relationship.find({"FeedId": "223"}, {userId: 1}) // Supp
3.    // The shardkey of the `feedInfo` collection is `userId`, and hashed presha
4.    db.FeedInfo.find({"userId": "3445"})
```

In short, you can use a secondary index collection to ultimately solve the problem of cross-shard broadcasting.

However, this approach increases the storage costs and adds a secondary index query. We recommend that you use it only in scenarios where the number of shards is high and queries without the shardkey are frequent.

Instance

Last updated：2024-01-15 14:49:56

How do I view the details of a TencentDB for MongoDB instance?

You can click an instance ID in the instance list to enter the details page and view the details.

How do I access a TencentDB for MongoDB instance?

TencentDB for MongoDB can be connected by using a variety of programming languages, such as Shell, PHP, Node.js, Java, and Python.

For more information, see Connecting to TencentDB for MongoDB Instance.

Is the time required for upgrading TencentDB for MongoDB instance specifications related to the used capacity of the instance?

The time required for upgrading an instance depends on its used capacity. Primary-secondary switch will occur once during the upgrade, and the instance will be temporarily inaccessible for about ten seconds.

How do I create a TencentDB for MongoDB instance?

On the purchase page, specify the specification and usage period and click **Buy Now** to create an instance.

How do I find a TencentDB for MongoDB instance assigned to a specific project?

To find an instance assigned to a specific project, call the DescribeDBInstances API to query the list of instances.

What is the number of connections to a TencentDB for MongoDB instance? Can I increase the number?

See Use Limits. The number of connections depends on instance specifications. You can increase the number by upgrading instance specifications.

How do I view the slow logs of a TencentDB for MongoDB instance?

You can log in to the console and select **Database Management** > **Slow Query Management** on the instance management page to query slow logs.

How do I query the instance specifications supported by TencentDB for MongoDB?

You can call the DescribeSpecInfo API to query the supported instance specifications.

How do I choose between a replica set instance and sharded cluster instance?

A **replica set** instance consists of one primary node and one or more secondary nodes. You can select a replica set instance in the following business scenarios:

The data volume does not exceed the total instance capacity.

You can check whether the **capacity** specification of the replica set instance selected during instance creation meets your requirements to select an appropriate instance.

The write traffic is low while the read traffic is high.

You can scale out secondary nodes to solve the performance bottleneck of excessive read traffic. For detailed directions, see Adjusting Instance Specification.

There are a large number of database collections, but the document data volume in each collection is low.

In scenarios where multiple randomly combined fields are queried, if a sharded cluster instance is used and it is hard to select an appropriate shard key index, we recommend that you use a replica set instance.

Compared with a replica set instance, a **sharded cluster** instance breaks through the bottlenecks of data capacity and write traffic and has a complete horizontal scalability. Select a sharded cluster instance in the following scenarios:

The estimated data volume exceeds the capacity limit of a replica set instance.

You can view the **capacity** limit of a replica set instance during instance creation to select an appropriate instance.

The write traffic exceeds the write traffic limit of a replica set instance.

How do I set the priority for database read operations?

If your business does not require a high data consistency, you can configure read/write separation to reduce the request pressure on the primary node and improve the read performance. By default, TencentDB for MongoDB reads the primary node first. If you want to specify access to secondary nodes, set the **Read Preference** parameter on the client to configure the read policy. The values of **Read Preference** are as detailed below:

Value	Description	Default
primary	Reads the primary node only.	Yes
primaryPreferred	Reads the primary node first. If it is not available, a secondary node will be read.	No
secondary	Reads a secondary node only. If it is not available, an error will be reported.	No
secondaryPreferred	Reads a secondary node first. If it is not available, the primary node will be read.	No
nearest	Reads the nearest node with the lowest network latency.	No

To read a secondary node first, you can configure the URI as follows:
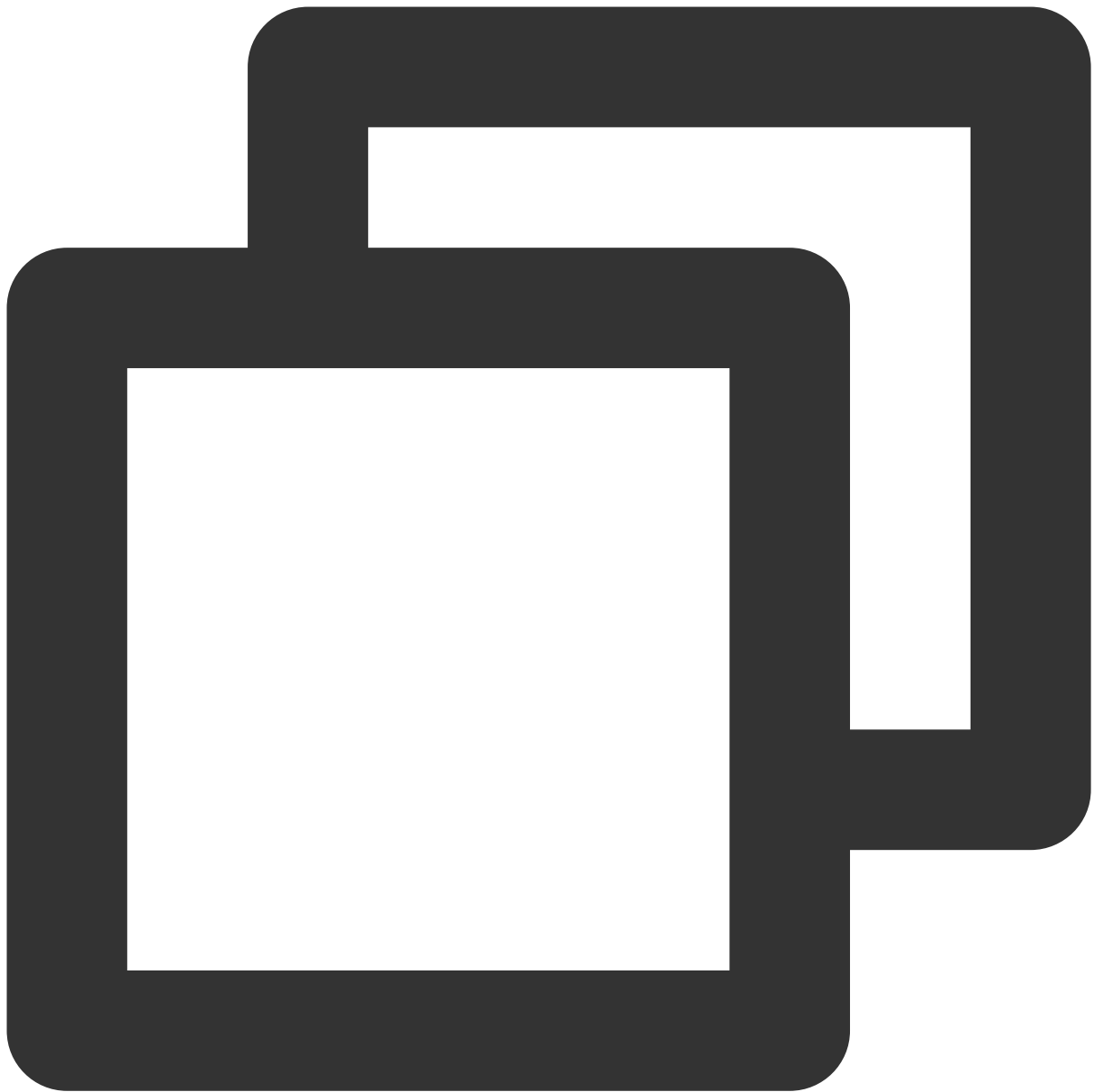
```
mongodb://username:password@IP:27017/admin?readPreference=secondaryPreferred
```

How do I solve the problem of reduced write performance when the majority of nodes are written?

Issue

For data writes, you can select different write concern policies based on your requirements for business data reliability.

Option	Description	Scenario

| {w: 0}	No acknowledgment messages are required for data writes by the client.	The data integrity is not concerned.
{w: 1}	An acknowledgment message will be sent to the client after data is written to the primary node. This is the default write concern option.	A balance between the performance and data reliability is required.
{w: majority}	An acknowledgment message will be sent to the client after data is written to the majority of nodes in the replica set.	A high data integrity is required in order to avoid data rollbacks.

If you requires a high data reliability, you can set the write concern option `w` to `majority` and use the `{j: true}` option to ensure that the acknowledgment message will be returned only after the journal log is persisted during data write. This can avoid data rollbacks but significantly decreases the write performance.

Optimization

You can optimize the write performance by disabling the chained replication feature. If there are three nodes A, B, and C, among which node B syncs data from node A and node C syncs data from node B, then a chain sync structure (A > B > C) is formed as shown below:
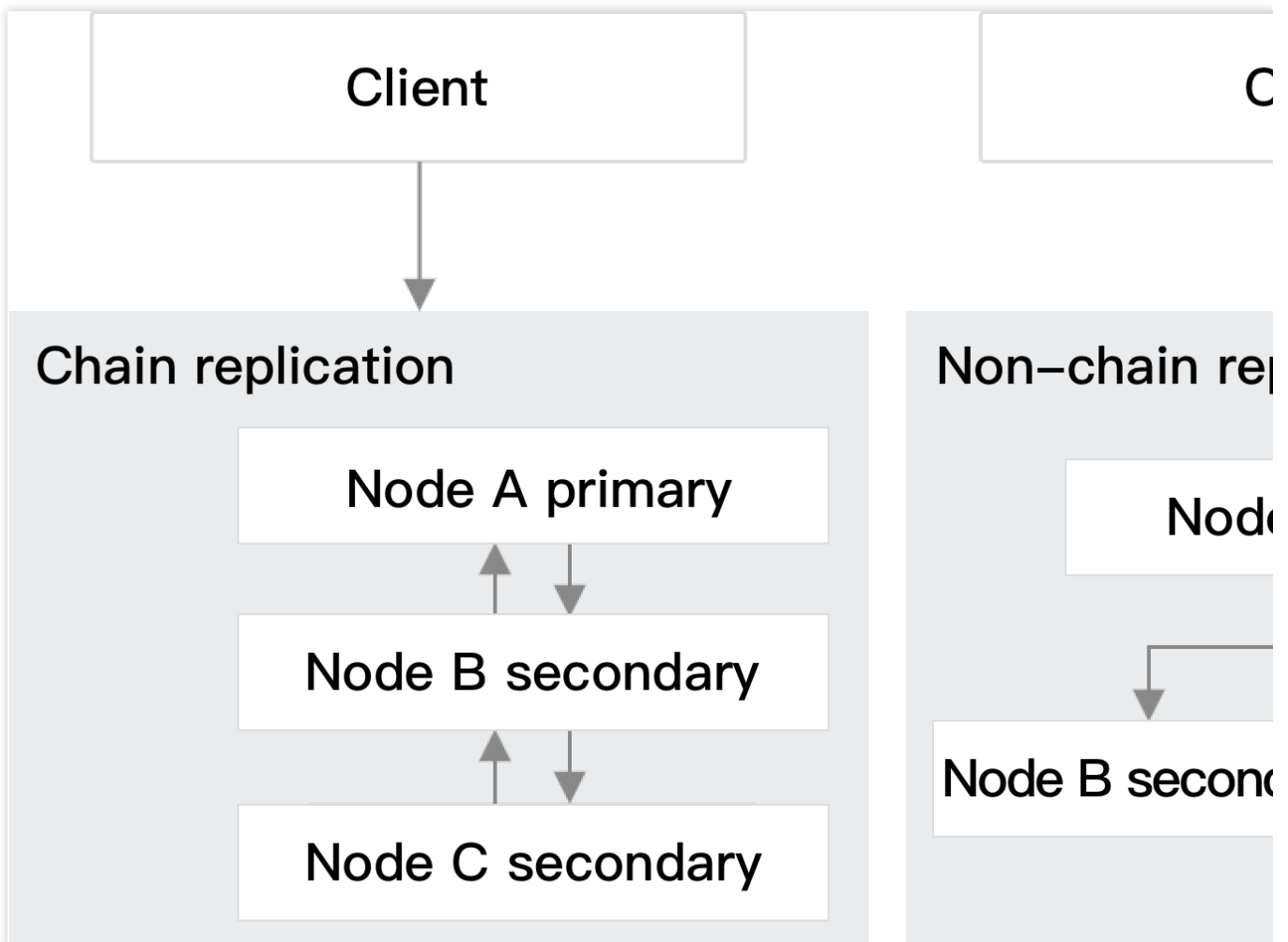
1. TencentDB for MongoDB multi-node replica set instances support chained replication. Run the following command to check whether the current replica set supports chained replication:
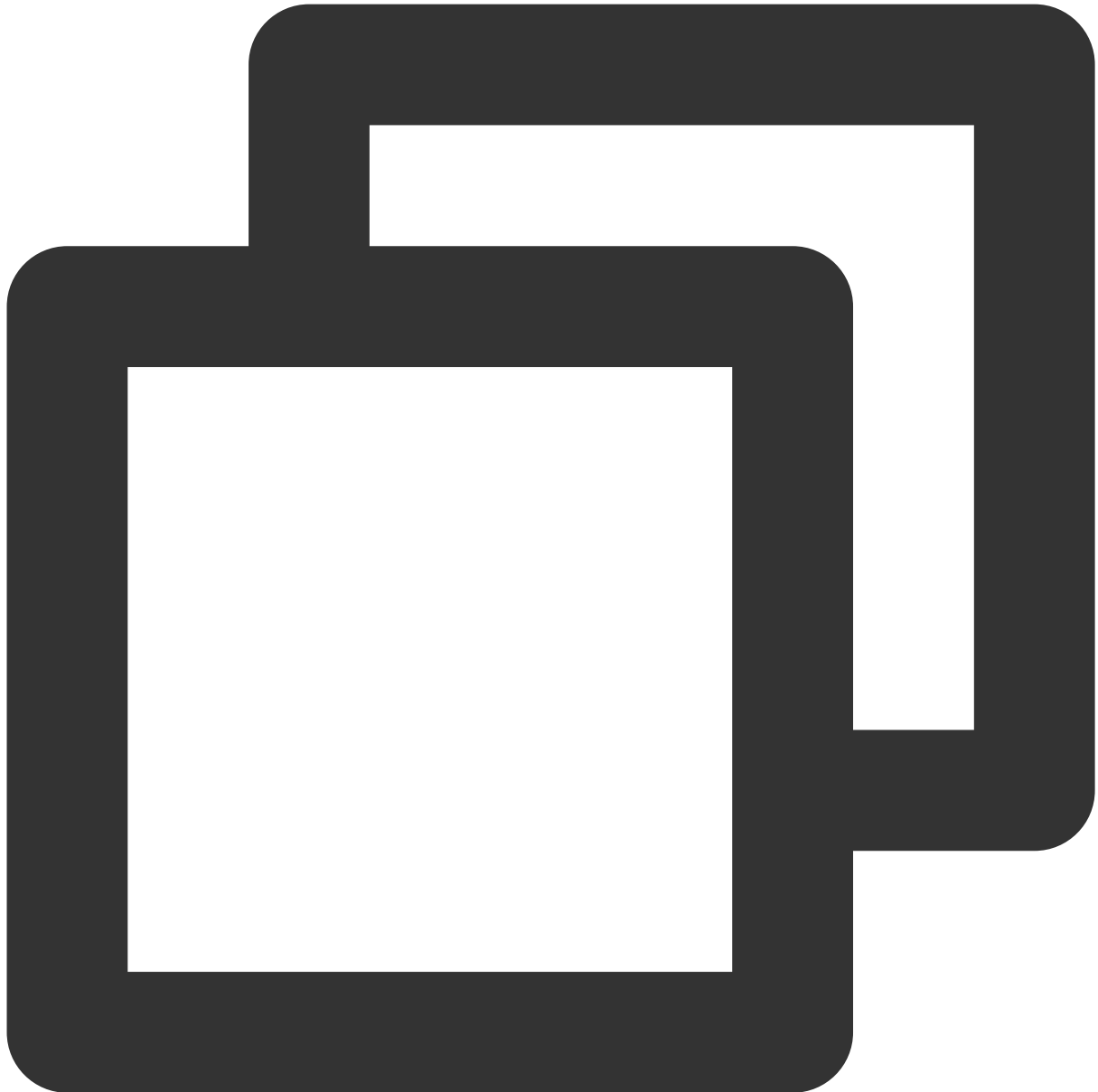

```
1.      cmgo-xx:SECONDARY> rs.conf().settings.chainingAllowed
2.      true
3.      cmgo-xx:SECONDARY>
```

2. Check whether chained replication exists in the current replica set nodes. You can view the sync source of each node. If a sync source is a secondary node, there is chained replication in the replica set. Below is the sample code:

```
1.      cmgo-xx:SECONDARY> rs.status().syncSourceHost
2.      xx.xx.xx.xx:7021
3.      cmgo-xx:SECONDARY>
```

3. Run the following command to disable chained replication:

```
1.      cfg = rs.config()
2.      cfg.settings.chainingAllowed = false
3.      rs.reconfig(cfg)
```

Rollback and Backup

Last updated：2024-01-15 14:49:56

How do I perform daily automatic backup and manual backup in TencentDB for MongoDB?

TencentDB for MongoDB supports two backup modes: daily automatic backup and manual backup. The backup data is retained for 7 days by default.

Automatic backup

An instance can be automatically backed up once a day. You can view details on the **Backup and Rollback** tab on the management page in the TencentDB for MongoDB console by clicking the instance ID.

Manual backup

On the **Backup and Rollback** tab, click **Manual Backup** in the top-right corner, enter remarks in the pop-up window, and click **Submit** to complete manual backup.

What determines the rollback time of a TencentDB for MongoDB instance?

Rollback is based on the latest full backup image and oplog. The rollback time is subject to the amount of replayed oplogs.

If the rollback is performed a long time after the full backup is completed, it will take more time to replay oplogs.

How do I back up and roll back an instance in TencentDB for MongoDB?

Replica set or sharded cluster instances currently support instance-level and database/collection-level backup and rollback.

Backup

In the TencentDB for MongoDB console, click an instance ID to enter the instance management page and click **Manual Backup** or set automatic backup on the **Backup and Rollback** tab. For detailed directions, see Data Backup.

Rollback

You need to enter a date to which you want to roll the instance back. You can enter any time point within the last 7 days. When you need to restore multiple collections of the database, you can perform **Collection Rollback** in the console to restore the data to the current instance or a new instance. When you need to restore the data of the entire instance, you can directly clone a new instance based on the backup file of the current instance to restore the data. For detailed directions, see Data Recovery.

How do I reduce jitters during instance backup?

By default, TencentDB for MongoDB performs full and incremental backups for the cluster data at midnight and supports rollback to any time point within the past 7 days. However, as the volume of the cluster data increases, the cluster may experience jitters regularly in the following aspects:

1. The access latency increases.

2. Slow logs increase.

3. The CPU utilization increases.

Analysis shows that the problem occurs when data backup is performed. Because the data of the entire instance needs to be backed up during physical and logical backups, the system resource load increases, which eventually affects the business query service.

Solution: Hide the node during data backup to ensure that it is invisible to users. You can submit a ticket for assistance.

Connection

Last updated：2024-01-15 14:49:55

What should I do in case of disconnection from TencentDB for MongoDB?

See Connecting to TencentDB for MongoDB Instance to eliminate authentication issues.

What should I do if the message "Remote server has closed the connection" is displayed in TencentDB for MongoDB?

First, see Connecting to TencentDB for MongoDB Instance to eliminate authentication issues. If you can connect to MongoDB but still encounter this problem, you may need a reconnection mechanism. For more information, see PHP Reconnection Sample.

WiredTiger 3.2 has the problem of table lock. Is TencentDB for MongoDB affected by the same issue?

It depends on the specific situation. For example, a global lock is required to create indexes by default, and a lock is also needed when you execute the `fsynclock` command.
Lock is a feature of databases, and is used to deal with the problems of concurrent access. Generally, a lock is required, as long as it does not affect the operation of your business.

Which version of the driver should be used in MongoDB?

We recommend that you use the latest version, for example, mongo-1.6 or later for PHP.

Which programming languages can be used to connect to MongoDB?

TencentDB for MongoDB can be connected by using a variety of programming languages, such as Shell, PHP, Node.js, Java, and Python. For more information, see Connecting to TencentDB for MongoDB Instance.

Which programming language clients are supported by TencentDB for MongoDB?

TencentDB for MongoDB provides the same compatibility as MongoDB. TencentDB for MongoDB supports any client that is supported by the official MongoDB. For example: C, C++, C#, Java, Node.js, Python, PHP, and Perl. For more information, see Start Developing with MongoDB.

How do I connect to TencentDB for MongoDB in shell?

For more information, see Shell Connection Sample.

What URIs in service applications can be used to connect to MongoDB?

For more information, see Connecting to TencentDB for MongoDB Instance.

What should I do if I cannot connect to TencentDB for MongoDB using meteor and other types of frameworks and class libraries?

This is usually caused by incorrect connection method or URI. Check and verify the cause first.

How do I set the maximum number of connections to MongoDB in PHP?

MongoDB driver can control the number of connections by configuring the `maxPoolSize` parameter in the URL connection.

In the MongoDB driver, you can set the connection count through the `Mongo::setPoolSize()` method.

What is the limit on the number of connections to MongoDB?

For more information, see Product Specifications.

How do I reconnect to MongoDB manually?

Instead of simply allowing you to access mongod, TencentDB for MongoDB database service provides a load balancer IP for access. You can use this IP to connect to a range of route access layers similar to mongos.
The client driver establishes a persistent connection with an access server using a load balancer IP. If the connection is active for a long period of time, Tencent Cloud will not impose an intervention on this status. However, if the persistent connection is inactive for more than one day (this period will be adjusted with optimized version), the route access layer will terminate the connection.
Generally, the client driver will implement an automatic reconnection. However, this process cannot be implemented by some language drivers. For the language drivers that cannot implement automatic reconnection, if you attempt to communicate with the TencentDB for MongoDB service using a terminated connection, an error message such as "Remote server has closed the connection" will be returned. So manual reconnection is required. Here is a demo for PHP reconnection.
Reconnection based on PHP mongo driver

```php
<?php

function getConnection() {
    $connection = false;
    $uri = 'mongodb://rwuser:1234567a@10.66.148.142:27017/admin?authMechanism=MONGODB-CR';
    $maxRetries = 5;
    for( $counts = 1; $counts <= $maxRetries; $counts++ ) {
        try {
            $connection = new MongoClient($uri);
        } catch( Exception $e ) {
        // 或者 根据需要使用下面的catch代码行，注意那一个"\"，某些框架使用命名空间时需要用到。
        // } catch( \Exception $e ) {
            continue;
        }
        break;
    }
    return $connection;
}

$connection = getConnection();

if($connection) {
        $db = $connection->testdb;
        $collection = $db->testcollection;

        $one = $collection->findOne();

        var_dump($one);
}
```
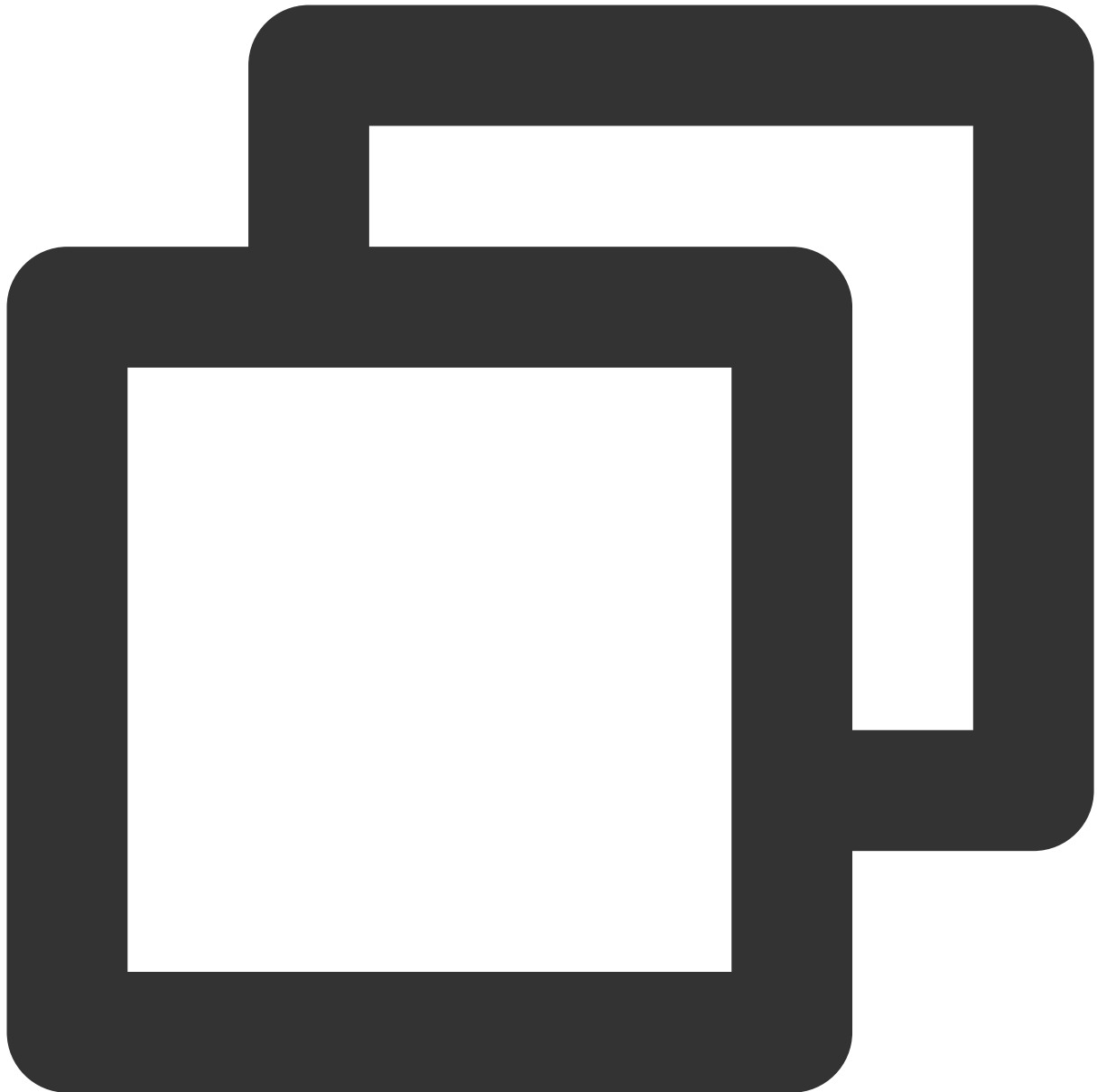
How do I use mongoose to connect to TencentDB for MongoDB?

Here are the parameters for mongoose to connect to TencentDB for MongoDB:

```
var dbUri = " mongodb:// " + user + " : " +password + " @ " +host + ":" +port + " /

var opts = {
    auth: {
        authMechanism : ' MONDODB-CR'
        }
};
var connection = mongodb.createConnection(dbUri, opts);
```

Can I connect to MongoDB from a public network?

TencentDB for MongoDB can only be connected over the private network. For more information, see Connecting to TencentDB for MongoDB Instance.

Connection through a public network is not supported. To connect to MongoDB locally, you can implement port forwarding using the server within the same private network under the same MongoDB account.

Data Migration

Last updated：2024-01-15 14:49:55

How do I set parameters to export data from TencentDB for MongoDB?

In mongodump parameters, set `--readPreference=secondaryPreferred` .

What data migration types are supported in TencentDB for MongoDB?

Migration from CVM-based self-created and public network-based instances is supported. For more information, please see TencentDB for MongoDB Data Migration.

How do I export MongoDB data to my local device by using mongodump (for an entire database) or mongoexport (for a single collection)?

In CVM, you can use the MongoDB shell client to connect to TencentDB for MongoDB for data export. **Please use the latest MongoDB client suite.**

MongoDB provides two sets of official tools for data export. Generally, mongodump is used to export an entire database, where the BSON data format is used to facilitate massive data dump, while mongoexport is used to export a single collection, where the JSON data format is used for higher readability.

1. Use mongodump to export an entire database for backup
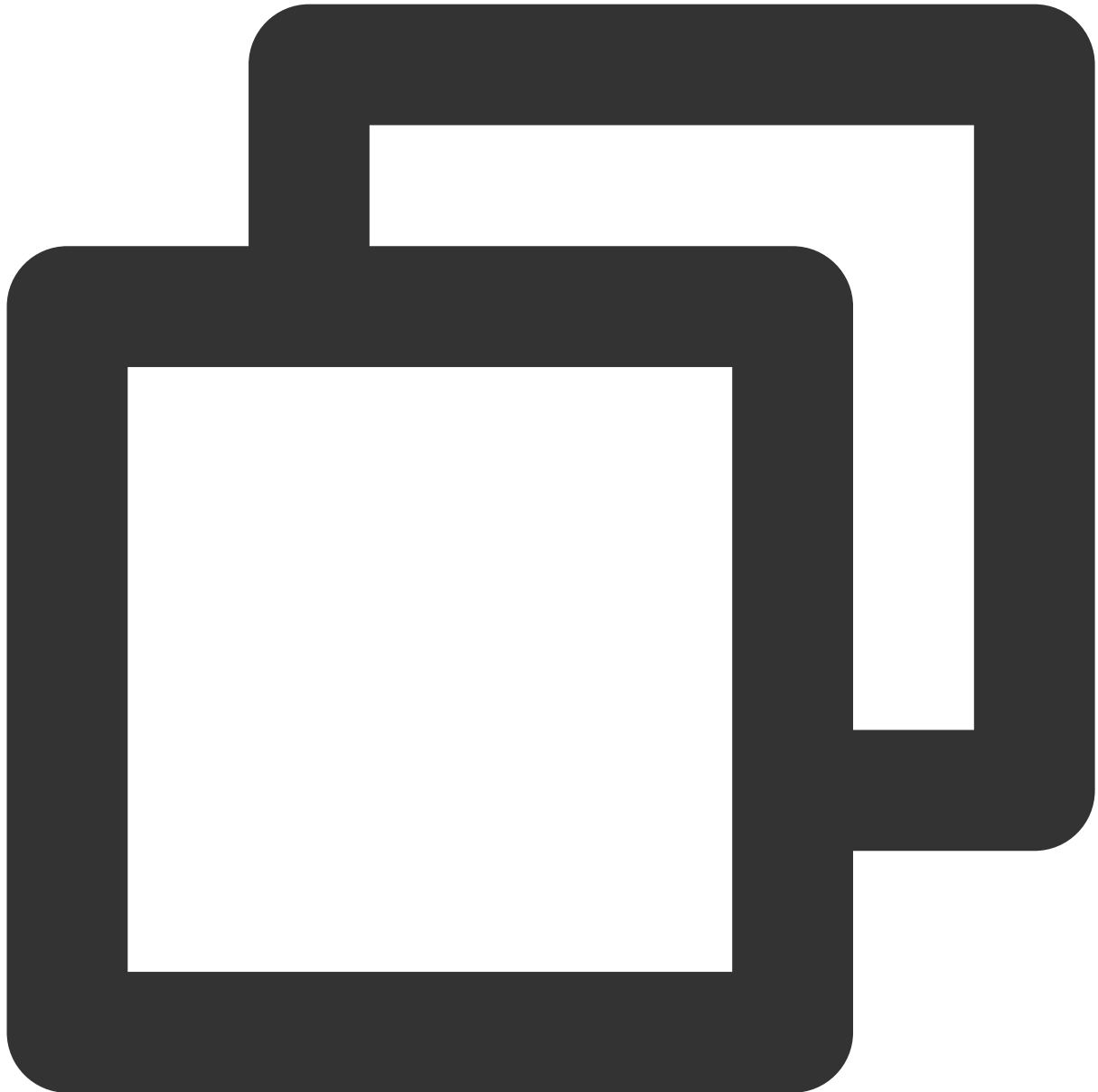
 The export command is as follows:

```
mongodump --host 10.66.187.127:27017 -u mongouser -p thepasswordA1 --authenticatio
```

```
#: ./mongodump --host 10.66.187.127:27017 -u mongouser -p thepasswordA1 --authenticationDatabase=admin
 --db=testdb -o /data/dump_testdb
2016-11-16T12:12:06.114+0800    writing testdb.system.indexes to
2016-11-16T12:12:06.116+0800    done dumping testdb.system.indexes (1 document)
2016-11-16T12:12:06.116+0800    writing testdb.testcollection to
2016-11-16T12:12:06.118+0800    done dumping testdb.testcollection (3 documents)
#:
```

2. Use mongoexport to export a single collection for backup

The export command is as follows:


```
mongoexport --host 10.66.187.127:27017 -u mongouser -p thepasswordA1 --authenticati
```

Note:

In addition, you can include the `-f` parameter to specify a desired field or the `-q` parameter to specify a query condition so as to restrict the data to be exported.

## 3. Parameters for export commands written by the users `rwuser` and `mongouser`

As described in the Connection Sample, TencentDB for MongoDB provides two usernames `rwuser` and

`mongouser` by default to support MONGODB-CR and SCRAM-SHA-1 authentication respectively.

For `mongouser` and all new users created in the console, simply follow the above samples to use the export tools.

For `rwuser` , the parameter `--authenticationMechanism=MONGODB-CR` should be included in each command.

mongodump sample:
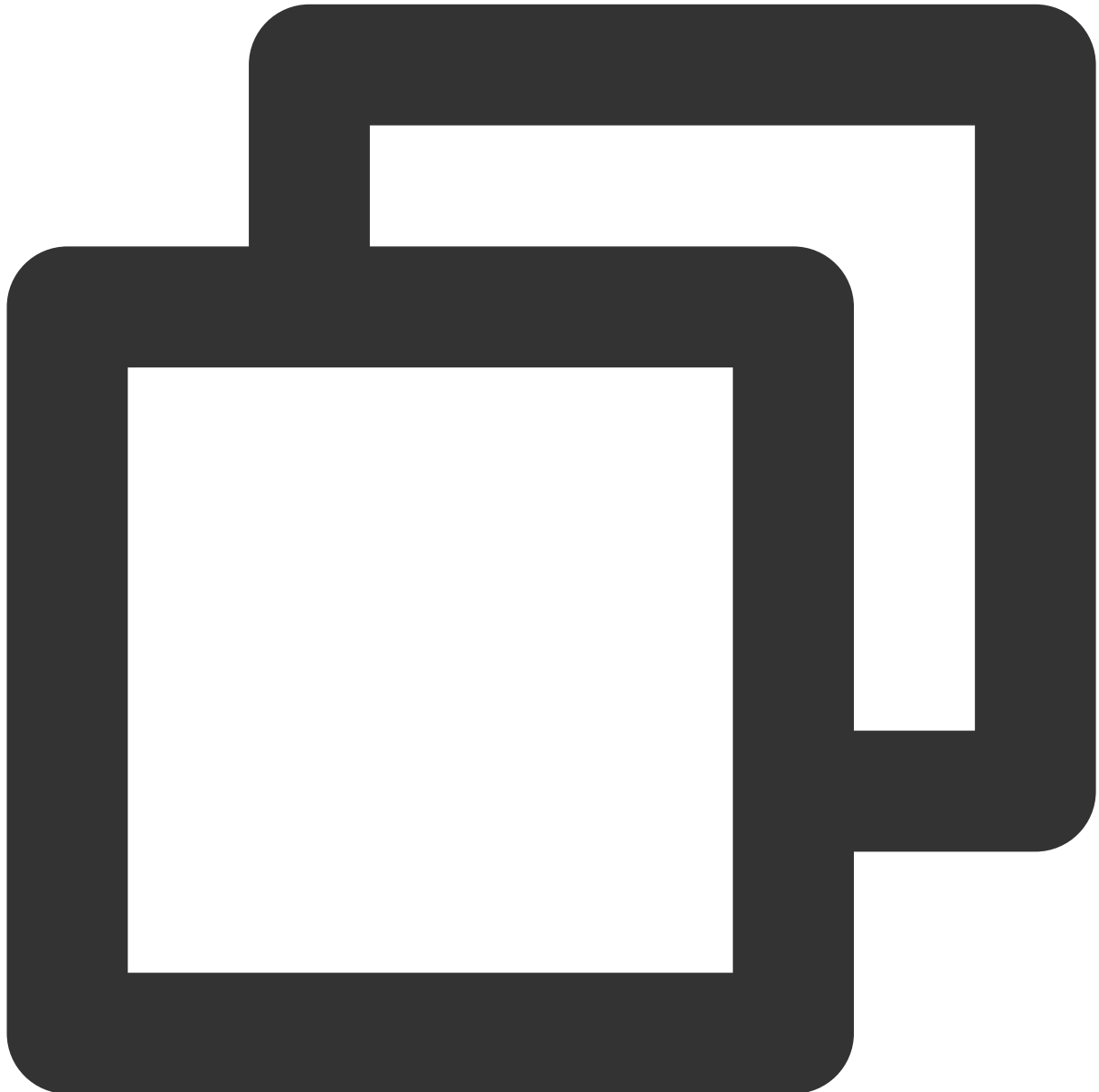
```
mongodump --host 10.66.187.127:27017 -u rwuser -p thepasswordA1 --authenticationDa
```

How do I import local data into TencentDB for MongoDB by using mongorestore (for an entire database) or mongoimport (for a single collection)?

In CVM, you can use the MongoDB shell client to connect to TencentDB for MongoDB for data import. **Please use the latest MongoDB client suite.**

MongoDB provides two sets of official tools for data import. Generally, mongorestore is used to import an entire database, where the BSON data format is used to facilitate massive data mongorestore, while mongoimport is used to import a single collection, where the JSON data format is used for higher readability.**1. Use mongorestore to import an entire database for backup**
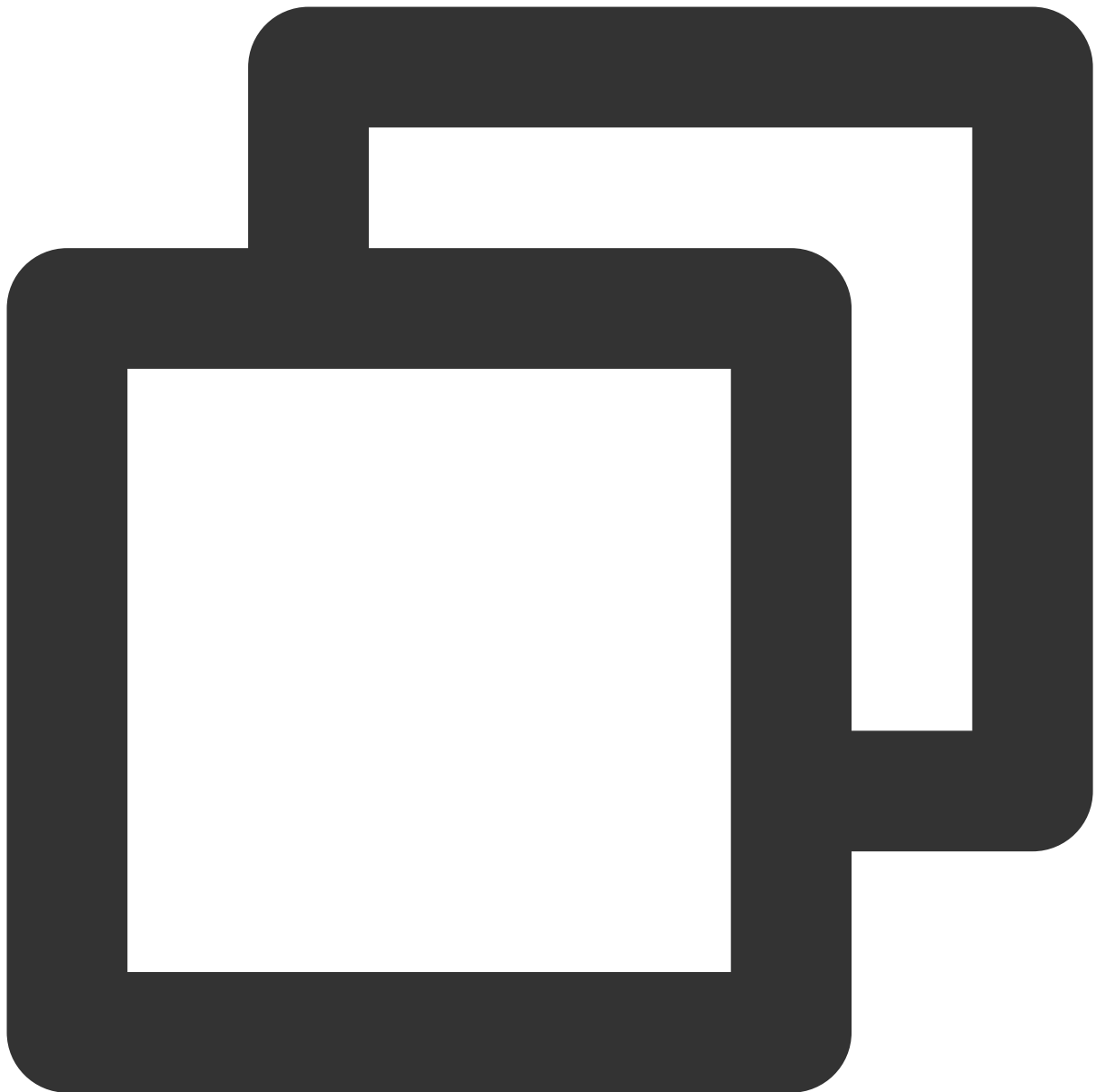
The import command is as follows:
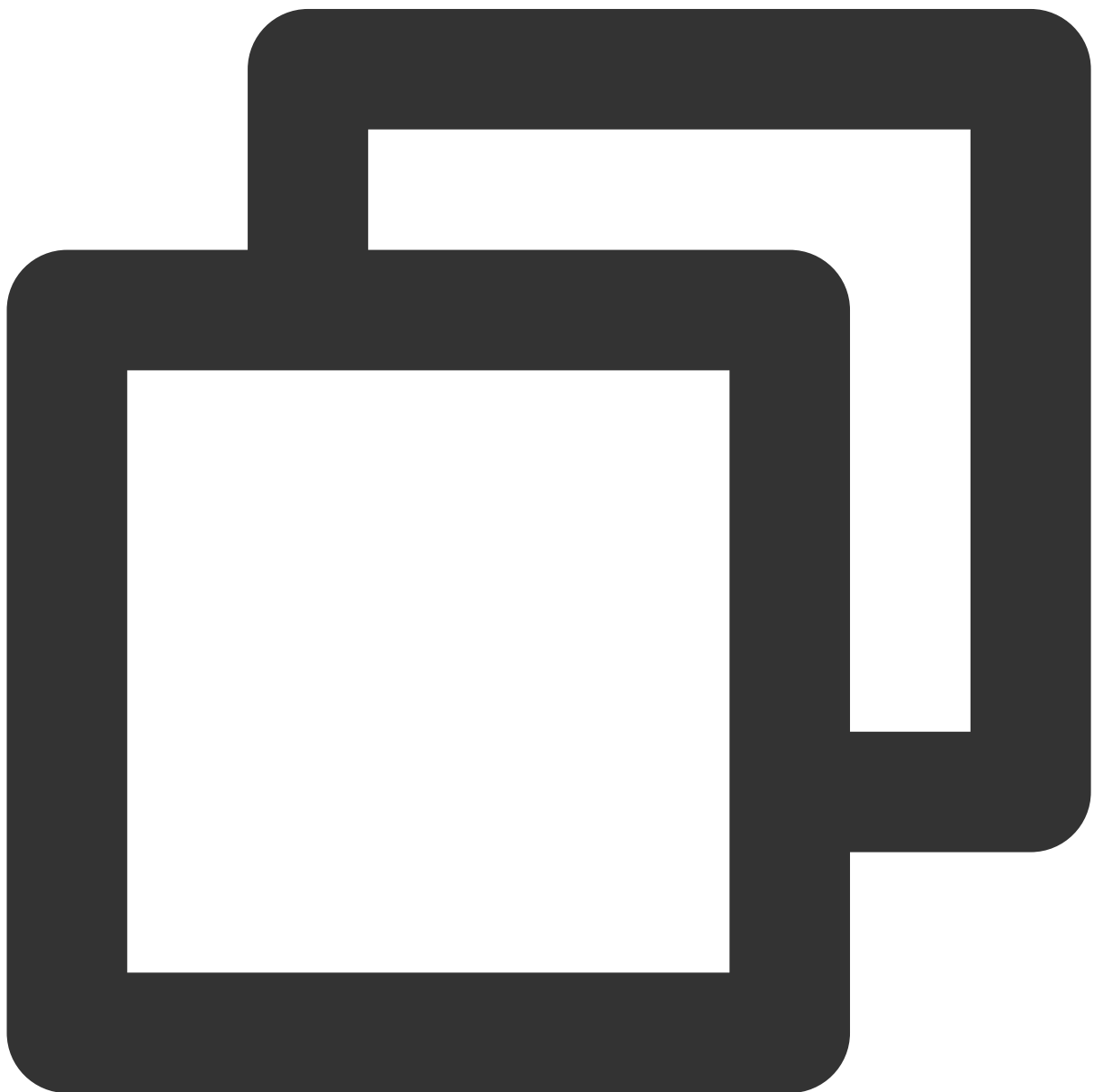
```
mongorestore --host 10.66.187.127:27017 -u mongouser -p thepasswordA1 --authenticat
```

```
#: ./mongorestore --host 10.66.187.127:27017 -u mongouser -p thepasswordA1 --authenticationDatabase=ad
min --dir=/data/dump_testdb
2016-11-16T12:13:23.654+0800    building a list of dbs and collections to restore from /data/dump_test
db dir
2016-11-16T12:13:23.678+0800    reading metadata for testdb.testcollection from /data/dump_testdb/test
db/testcollection.metadata.json
2016-11-16T12:13:23.678+0800    restoring testdb.testcollection from /data/dump_testdb/testdb/testcoll
ection.bson
2016-11-16T12:13:23.740+0800    restoring indexes for collection testdb.testcollection from metadata
2016-11-16T12:13:23.740+0800    finished restoring testdb.testcollection (3 documents)
2016-11-16T12:13:23.741+0800    done
#:
```

2. Use mongoimport to import a single collection for backup

The import command is as follows:

```
mongoimport --host 10.66.187.127:27017 -u mongouser -p thepasswordA1 --authenticati
```

3. Parameters for import commands written by the users `rwuser` and `mongouser`

As described in the Connection Sample, TencentDB for MongoDB provides two usernames `rwuser` and `mongouser` by default to support MONGODB-CR and SCRAM-SHA-1 authentication respectively.

For `mongouser` and all new users created in the console, simply follow the above samples to use the import tools.

For `rwuser`, the parameter `--authenticationMechanism=MONGODB-CR` should be included in each command.

mongorestore sample:
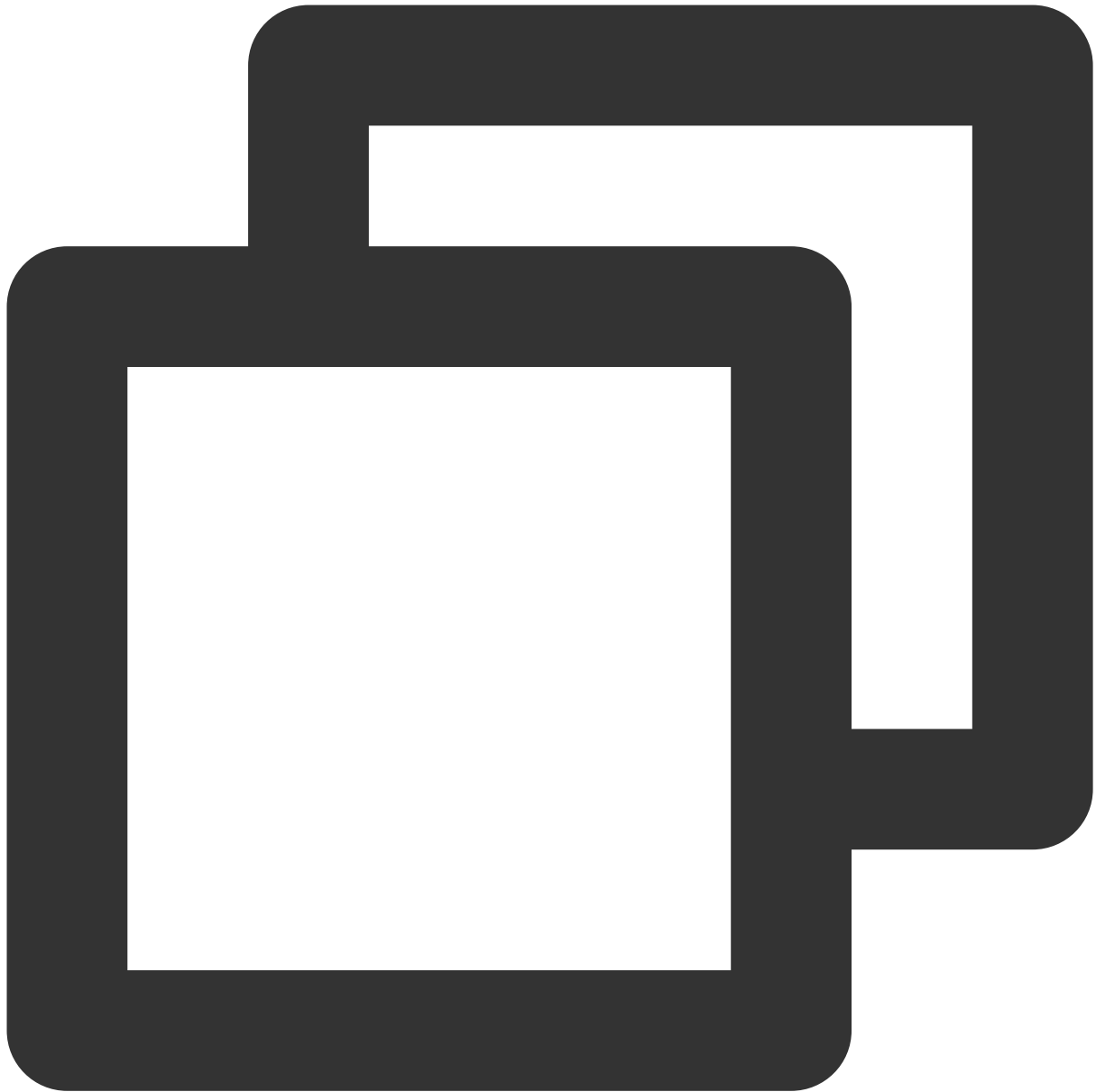
```
mongorestore --host 10.66.187.127:27017 -u rwuser -p thepasswordA1 --authenticatio
```

Why does the data imported into TencentDB for MongoDB occupy less capacity than in self-created MongoDB?

Possible causes:

There are a large number of creation, deletion, and update operations accumulated after the database has been

running for a long time.

MongoDB allocates a larger capacity than the actual data size during write operations in order to improve performance.

The original capacity is not reused after the data is deleted.

All these result in an overall higher void rate in the entire database capacity. Meanwhile, importing data can be considered as an operation similar to disk defragging, which makes imported data more compact and appear smaller in size.

What should I do if mongodump is unable to export data from MongoDB?

For more information on how to use mongodump, please see Import and Export. You are recommended to use mongodump 3.2.10 or above.

Others

Last updated：2024-01-15 14:49:55

What is the latest version of TencentDB for MongoDB?

Currently, TencentDB for MongoDB 3.2, 3.6, 4.0, and 4.2 are available.

How do I delete a TencentDB for MongoDB instance? Will it be deleted automatically if I don't renew it after it expires?

A TencentDB for MongoDB instance will be automatically terminated if it is not renewed after expiration. Confirm and back up your data promptly. You can also actively terminate it by selecting **More** > **Return and Refund**/**Terminate** in its **Operation** column in the instance list in the console.
For more information, see Terminating Instance.

How do I apply for security credentials for TencentDB for MongoDB?

Before calling a TencentCloud API for the first time, you need to apply for security credentials in the Tencent Cloud console.
Security credentials consist of a `SecretId` and a `SecretKey` :
`SecretId` is used to identify the API caller.
`SecretKey` is used to encrypt the strings to create a signature so that Tencent Cloud server can verify the identity of the caller.
Note:
API key is very important for creating TencentCloud API requests. With TencentCloud APIs, you can access and manage the resources in your Tencent Cloud account. For security reasons, store your keys safely and rotate them regularly (be sure to delete the old key when a new one is created). For more information, see Introduction.

What are the restrictions on using the MongoDB username?

TencentDB for MongoDB comes with a default user: "mongouser". It supports the SCRAM-SHA-1 authentication mechanism, and its role is readWriteAnyDatabase+dbAdmin. You can use it to read from and write to any database except for performing high-risk operations.
TencentDB for MongoDB 3.2 supports another default user: "rwuser". It uses the MONGODB-CR authentication mechanism which, however, has been no longer supported by MongoDB. We recommend that you use the "mongouser" to connect to your instance.
You can also manage your account and permissions as needed in the TencentDB for MongoDB console.