Document Database Service

User Guide

Issue 01

Date 2025-12-29

HUAWEI TECHNOLOGIES CO., LTD.

Trademarks and Permissions


and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.
All other trademarks and trade names mentioned in this document are the property of their respective holders.

Notice

The purchased products, services and features are stipulated by the contract made between Huawei and the customer. All or part of the products, services and features described in this document may not be within the purchase scope or the usage scope. Unless otherwise specified in the contract, all statements, information, and recommendations in this document are provided "AS IS" without warranties, guarantees or representations of any kind, either express or implied.

The information in this document is subject to change without notice. Every effort has been made in the preparation of this document to ensure accuracy of the contents, but all statements, information, and recommendations in this document do not constitute a warranty of any kind, express or implied.

Security Declaration

Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process.* For details about this process, visit the following web page:
https://www.huawei.com/en/psirt/vul-response-process
For vulnerability information, enterprise customers can visit the following web page:
https://securitybulletin.huawei.com/enterprise/en/security-advisory

Contents

1 Using IAM to Grant Access to DDS

1.1 Using IAM Roles or Policies to Grant Access to DDS

lets you manage of your DDS instances. With IAM, you can:

- Create IAM users for personnel based on your enterprise's organizational structure. Each IAM user has their own identity credentials for accessing DDS resources.
- Grant users only the permissions required to perform a given task based on their job responsibilities.
- Entrust or a cloud service to perform efficient O&M on your DDS resources.

If your account meets your permissions requirements, you can skip this section.

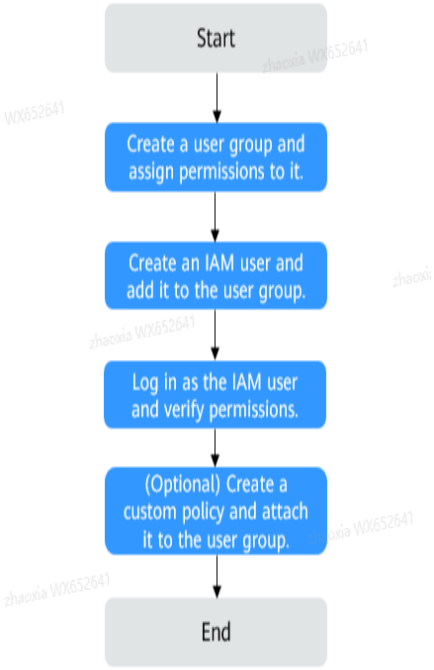
Figure 1-1 shows the process flow of role/policy-based authorization.

Prerequisites

Before granting permissions to user groups, learn about DDS system-defined permissions in . To grant permissions for other services, learn about all supported by IAM.

Process Flow

Figure 1-1 Process of granting DDS permissions using role/policy-based authorization



1. On the IAM console, (**DDS ReadOnlyAccess** as an example).
2. created in **1**.
3. and verify permissions.

   In the authorized region, perform the following operations:

   - Choose **Service List** > **Document Database Service**. Then click **Buy DB Instance** on the DDS console. If a message appears indicating that you have insufficient permissions to perform the operation, the **DDS ReadOnlyAccess** policy is in effect.
   - Choose another service from **Service List**. If a message appears indicating that you have insufficient permissions to access the service, the **DDS ReadOnlyAccess** policy is in effect.

Example Custom Policies

You can create custom policies to supplement the system-defined policies of DDS. For details about actions supported in custom policies, see .

To create a custom policy, choose either visual editor or JSON.

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

For details, see . The following lists examples of common DDS custom policies.

- Example 1: Grant permission to create DDS instances.

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "dds:instance:create"
         ]
      }
   ]
}
```

- Example 2: Grant permission to deny DDS instance deletion.

  A policy with only "Deny" permissions must be used together with other policies. If the permissions granted to an IAM user contain both "Allow" and "Deny", the "Deny" permissions take precedence over the "Allow" permissions.

  Assume that you want to grant the permissions of the **DDS FullAccess** policy to a user but want to prevent them from deleting DDS instances. You can create a custom policy for denying DDS instance deletion, and attach this policy together with the **DDS FullAccess** policy to the user. As an explicit deny in any policy overrides any allows, the user can perform all operations on DDS instances excepting deleting them. Example policy denying DDS instance deletion:

```
{
   "Version": "1.1",
   "Statement": [
      {
        "Effect": "Deny"
        "Action": [
            "dds:instance:deleteInstance"
         ],
       }
   ]
}
```

- Example 3: Create a custom policy containing multiple actions.

  A custom policy can contain the actions of one or multiple services that are of the same type (global or project-level). Example policy containing multiple actions:

```
{
      "Version": "1.1",
      "Statement": [
            {
                  "Action": [
                        "dds:instance:create",
                        "dds:instance:modify",
                        "dds:instance:deleteInstance",
             "vpc:publicIps:list",
             "vpc:publicIps:update"
                  ],
                  "Effect": "Allow"
            }
      ]
}
```

- Example 4: Configure a resource policy.

A custom policy can be configured with resources, indicating which resources the current action can operate on. Currently, the instance name can be configured. The asterisk (*) can be used as a wildcard. Example policy authorizing resources

```
{
   "Version": "1.1",
   "Statement": [
      {
         "Effect": "Allow",
         "Action": [
            "dds:instance:list"
         ]
      },
      {
         "Effect": "Allow",
         "Action": [
            "dds:instance:modify"
         ],
         "Resource": [
            "DDS:*:*:instanceName:dds-*"
         ]
      }
   ]
}
```

DDS Resources

A resource is an object that exists within a service. DDS resources include instances. To select these resources, specify their paths.

Table 1-1 DDS resources and their paths

Resource	Path
instance	[Format] dds:<region>:<account-id>:instanceName:<instance-name> [Notes] For instance resources, DDS automatically generates the prefix (**dds:*:*:instanceName:**) for resource paths. To specify a path for a specific instance, add the instance name to the end. To specify paths for multiple instances, add different instance names for each path. You can also use an asterisk * to indicate any instance. Example: **dds:*:*:instanceName:*** indicates any DDS instance.

DDS Request Conditions

Request conditions are useful in determining when a custom policy is in effect. A request condition consists of condition keys and operators. Condition keys are either global or service-specific and are used in the Condition element of a policy statement. (starting with **g:**) are available for operations of all services, while service-specific condition keys (starting with a service name such as **dds**:) are

available only for operations of specific services. An operator must be used together with a condition key to form a complete condition statement.

DDS has a group of condition keys predefined in IAM. For example, you can use **hw:SourceIp** to filter specified requesters' IP addresses and then allow actions. The following table lists the DDS predefined condition keys.

Table 1-2 DDS predefined condition keys

Condition Key	Operator	Description
dds:Encrypted	boolean	Filters access by the tag key that specifies whether to enable disk encryption in the request.
dds:BackupEnabled	boolean	Filters access by the tag key that specifies whether to enable the backup policy in the request.

1.2 Using IAM Identity Policies to Grant Access to DDS

System-defined permissions in provided by let you control access to DDS. With IAM, you can:

- Create IAM users or user groups for personnel based on your enterprise's organizational structure. Each IAM user has their own identity credentials for accessing DDS resources.
- Grant users only the permissions required to perform a given task based on their job responsibilities.
- Entrust or a cloud service to perform efficient O&M on your DDS resources.

If your account meets your permissions requirements, you can skip this section.

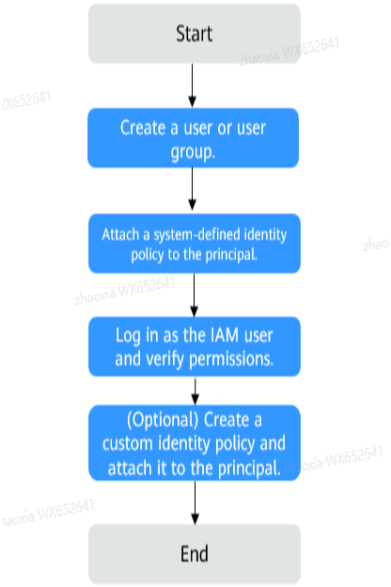
Figure 1-2 shows the process flow of identity policy-based authorization.

Prerequisites

Before granting permissions, learn about system-defined identity policies supported by DDS. For details, see . To grant permissions for other services, learn about all supported by IAM.

Process Flow

Figure 1-2 Process of granting DDS permissions using identity policy-based authorization



1. On the IAM console, .
2. (**DDSReadOnlyPolicy** as an example) to the user or user group.
3. and verify permissions.

   In the authorized region, perform the following operations:

   - Choose **Service List** > **Document Database Service**. Then click **Buy DB Instance** on the DDS console. If a message appears indicating that you have insufficient permissions to perform the operation, the **DDSReadOnlyPolicy** policy is in effect.
   - Choose another service from **Service List**. If a message appears indicating that you have insufficient permissions to access the service, the **DDSReadOnlyPolicy** policy is in effect.

Example Custom Identity Policies

You can create custom identity policies to supplement the system-defined identity policies of DDS. For details about actions supported in custom identity policies, see .

To create a custom identity policy, choose either visual editor or JSON.

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

For details, see .

When creating a custom identity policy, use the Resource element to specify the resources the identity policy applies to and use the Condition element (service-specific condition keys) to control when the identity policy is in effect. For details about the supported resource types and condition keys, see . The following lists examples of common DDS custom identity policies.

- Example 1: Grant permission to create and delete DDS instances.

```
{
    "Version": "5.0",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cbr:vaults:create",
                "cbr:vaults:delete"
            ]
        }
    ]
}
```

- Example 2: Create a custom identity policy containing multiple actions.

  A custom identity policy can contain the actions of one or more services. Example identity policy containing multiple actions:

```
{
    "Version": "5.0",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "cbr:vaults:create",
                "cbr:vaults:delete"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "evs:volumes:create",
                "evs:volumes:list"
            ]
        },
        {
            "Effect": "Allow",
            "Action": [
                "ecs:cloudServers:createServers",
                "ecs:cloudServers:listServersDetails"
            ]
        }
    ]
}
```


2 Connection Management

2.1 Configuring Cross-CIDR Access

Scenarios

When a replica set instance is connected through an internal network, a replica set node is configured with a management NIC (for receiving management instructions and internal communications of the instance) and a data NIC (for receiving and responding to service requests from the client), and the mapping between management IP addresses and data IP addresses of three standard CIDR blocks is configured by default.

- If your client and the replica set instance are deployed in different CIDR blocks and the client CIDR block is 192.168.0.0/16, 172.16.0.0/12, or 10.0.0.0/8, you do not need to configure access across CIDR blocks for the instance.
- If your client and the replica set instance are deployed in different CIDR blocks and the client CIDR block is not 192.168.0.0/16, 172.16.0.0/12, or 10.0.0.0/8, you must configure access across CIDR blocks for the instance to communicate with your client.
- No standard CIDR blocks are configured for replica set instances created before August 2025. If your client and the replica set instance are deployed in different CIDR blocks, you must configure access across CIDR blocks.

This section describes how to configure access across CIDR blocks for an instance.

Precautions

- Only replica set instances support this function.
- During the configuration of cross-CIDR access, services are running properly without interruption or intermittent disconnection.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

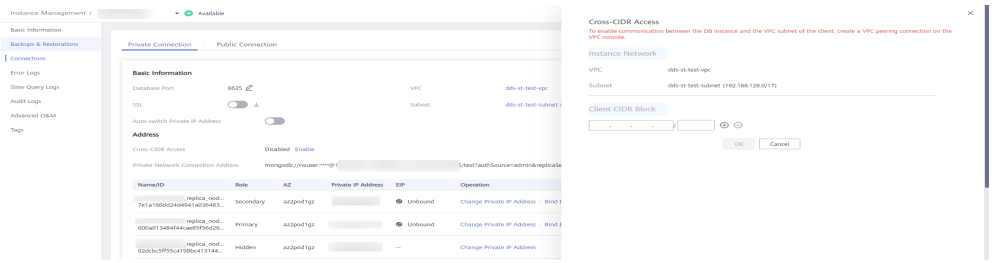
Step 4 On the **Instances** page, click the instance name.

Step 5 In the navigation pane on the left, choose **Connections**.

Step 6 On the **Private Connection** tab, click **Enable** to the right of **Cross-CIDR Access**. You can add or delete the blocks as required.

- Click to add new CIDR blocks.
- Click to delete existing CIDR blocks.

Figure 2-1 Cross-CIDR Access



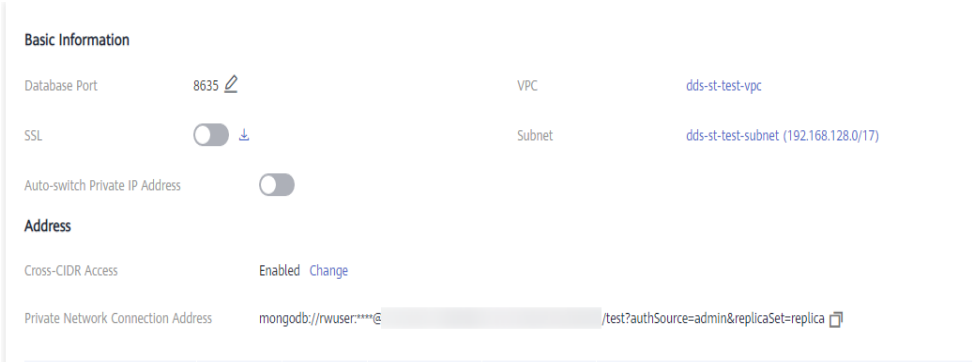
NOTE

Up to 30 CIDR blocks can be configured, and each of them can overlap but they cannot be the same. That is, the source CIDR blocks can overlap but cannot be the same. The CIDR blocks cannot start with 127. The allowed IP mask ranges from 8 to 32.

Step 7 View the change results. After cross-CIDR access is enabled, **Enabled** is displayed to the right of **Cross-CIDR Access**.

If you need to change the client CIDR block, click **Change** to the right of **Cross-CIDR Access**.

Figure 2-2 Changing a CIDR block



----End

Follow-up Operations

After cross-CIDR access is configured, you can use MongoShell to connect to a replica set instance over a private network. For details, see **Connecting to a Cluster Instance Using Mongo Shell**.

2.2 Enabling IP Addresses of Shard and Config Nodes

Scenarios

A cluster instance of Community Edition consists of dds mongos, shard, and config nodes. When your services need to read and write data from and into databases, connect to the dds mongos node. In certain scenarios (for example, data migration and synchronization between clusters), you need to read data from the shard or config node and will need to obtain the IP address of the corresponding node.

This section describes how to obtain the IP addresses of the shard and config nodes.

Before You Start

- DDS supports cluster instances of Community Edition 3.4, 4.0 and 4.2.
- DDS creates two connection addresses for the primary and secondary nodes in a shard group or config group.
- The network type of the connection address is the same as that of the current dds mongos node.
- Once the shard or config IP addresses are assigned to your nodes, they cannot be changed.
- If IPv6 is enabled in a subnet, you cannot enable IP addresses of the shard and config nodes for DB instances created using the subnet.
- After the shard IP address is enabled, restart the corresponding shard node for the configuration to take effect.
- After you enable the connection address, you can **connect to an instance using Mongo Shell**.

Enabling shard IP Address

NOTE

- The button for showing shard IP address can only be enabled. It cannot be modified.
- Once the shard IP address is enabled, DDS automatically applies for connection addresses for all shard nodes in the current instance.
- After the shard IP address is enabled and new shard nodes are added, you need to manually locate a newly added shard node and choose **More** > **Show shard IP Address** in the **Operation** column to show the shard IP address.
- After the shard IP address is enabled, the database user **sharduser** is created. For details about how to reset the password, see **Resetting the Password of User sharduser**.

Step 1 Log in to the management console.

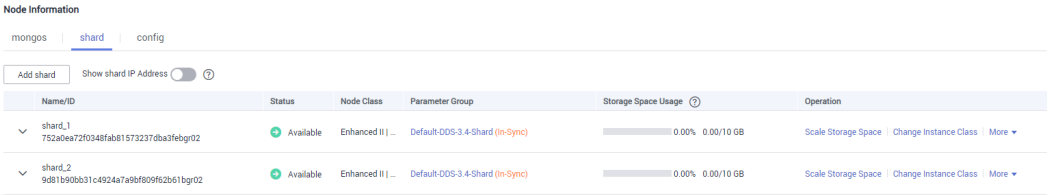
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name. The **Basic Information** page is displayed.

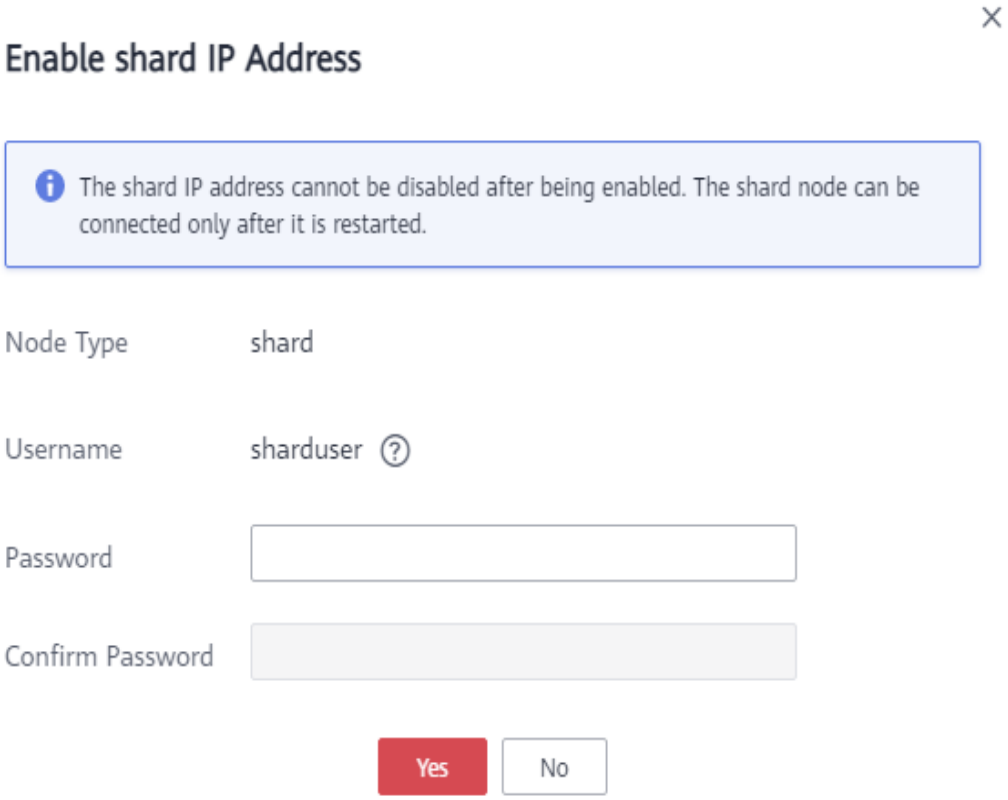
Step 5 In the **Node Information** area, click the **shard** tab.

Figure 2-3 shard nodes



Step 6 Click **Show shard IP Address**. In the displayed dialog box, enter and confirm the password for connecting to the node.

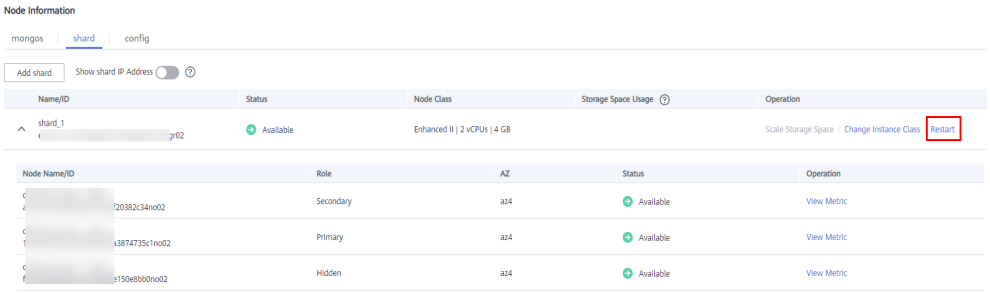
Figure 2-4 Enable shard IP Address



After the shard IP address is enabled, restart the corresponding shard node for the configuration to take effect.

In the **Node Information** area, locate the row that contains the shard node and click **Restart** in the **Operation** column to restart the shard node.

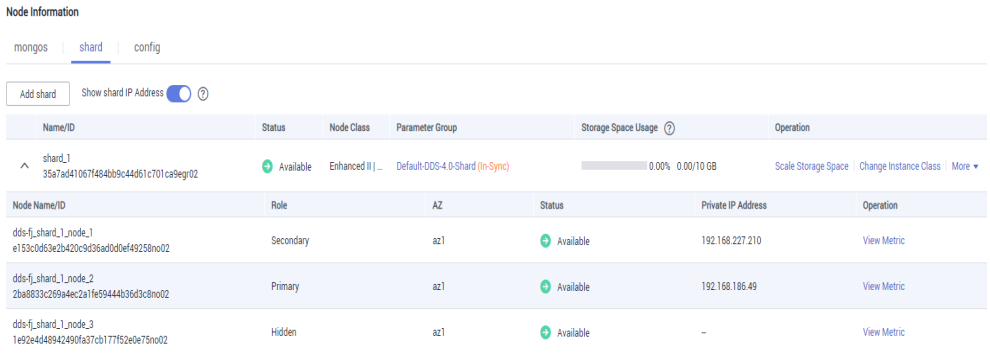
Figure 2-5 Restarting a shard node



Step 7 View the private IP address of the shard node.

After the shard IP address is enabled, you can click next to a shard node on the current page to expand the node drop-down list or click **Connections** in the navigation pane on the left, and then obtain the private IP address.

Figure 2-6 Private IP addresses of shard nodes



The connection address of the current shard node is as follows:

mongodb://***sharduser:***<password>***@192.168.xx.xx:8637,***192.168.xx.xx***:8637***/test?authSource=admin&replicaSet=shard_?

NOTE

- **sharduser** is the username of the current shard node.
- ******** is the password of the current node.
- **192.168.xx.xx** and **192.168.xx.xx** are the private IP addresses of the primary and secondary shard nodes.
- **8637** is the port of the shard node and cannot be changed.
- **shard_?** is the name of the shard node to be connected, for example, **shard_1**.

----End

Resetting the Password of User sharduser

NOTE

This function is available only after the shard IP address is enabled.

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

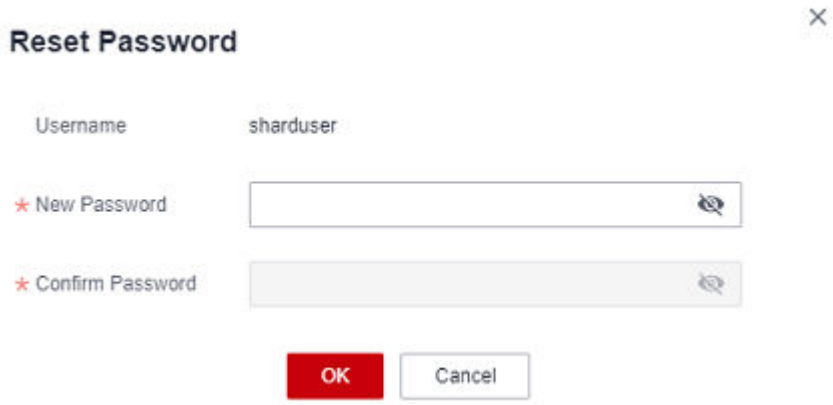
Step 4 On the **Instances** page, click the instance name. The **Basic Information** page is displayed.

Step 5 In the **Node Information** area, click the **shard** tab.

Figure 2-7 shard nodes

Step 6 Click **Reset Password**.

Figure 2-8 Resetting a password



Step 7 Enter the new password and click **OK**.

----End

Enabling config IP Address

NOTE

- The button for showing config IP address can only be enabled. It cannot be modified.
- Once the config IP address is enabled, DDS automatically applies for connection addresses for all config nodes in the current instance.
- After the config IP address is enabled, the database user **csuser** is created. For details about how to reset the password, see **Resetting the Password of User csuser**.

Step 1 Log in to the management console.

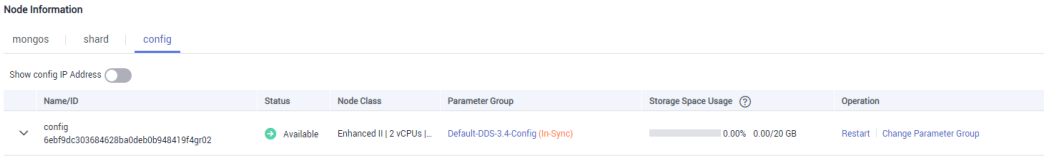
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the left navigation pane, choose **Instances**. In the instance list, click the instance name to go to the **Basic Information** page.

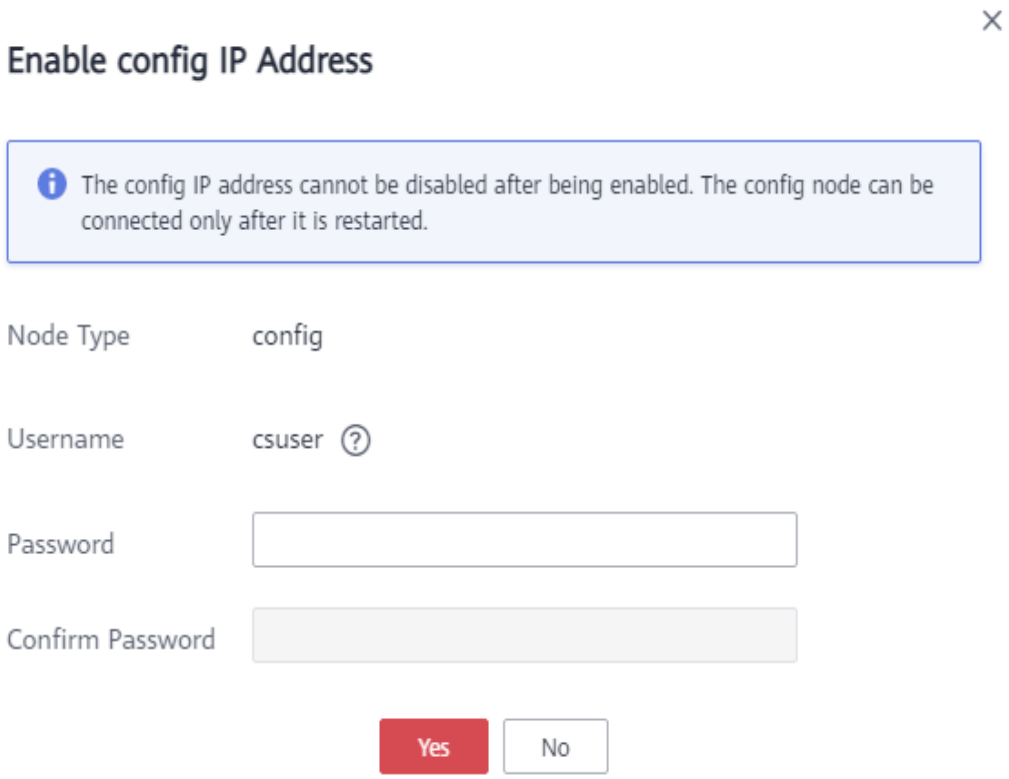
Step 5 In the **Node Information** area, click the **config** tab.

Figure 2-9 config nodes



Step 6 Click **Show config IP Address**. In the displayed dialog box, enter and confirm the password for connecting to the node.

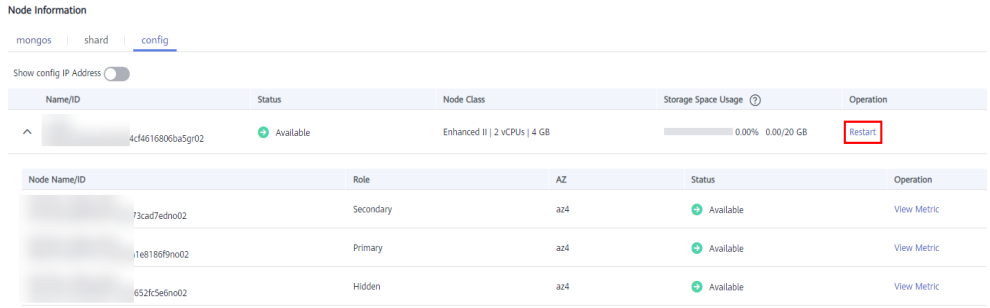
Figure 2-10 Enable config IP Address



After the config IP address is enabled, the corresponding config node needs to be restarted for the configuration to take effect.

In the **Node Information** area, locate the row that contains the config node and click **Restart** in the **Operation** column to restart the config node.

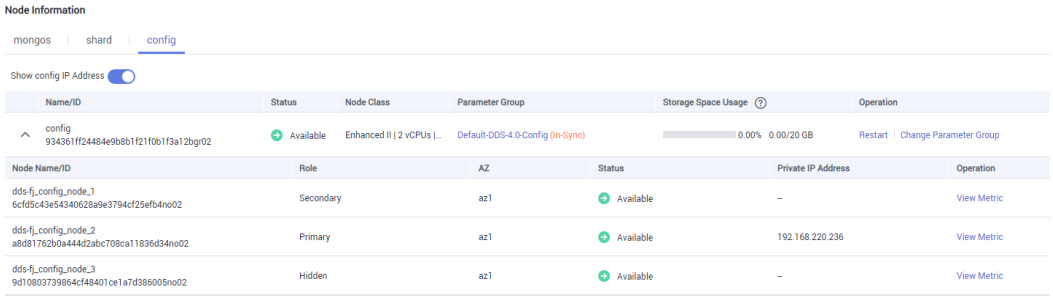
Figure 2-11 Restarting a config node



Step 7 View the private IP address of the config node.

After the config IP address is enabled, you can click next to the node on the current page to expand the node drop-down list or click **Connections** in the navigation pane on the left, and then obtain the private IP address.

Figure 2-12 Private IP addresses of config nodes



The connection address of the current config node is as follows:

mongodb://***csuser:***<*password*>***@192.168.xx.xx:8636***/test?authSource=admin

NOTE

- **csuser** is the username of the current config node.
- ******** is the password of the current node.
- **192.168.xx.xx** is the private IP address of the primary config node.
- **8636** is the port of the config node and cannot be changed.

----End

Resetting the Password of User csuser

NOTE

This function is available only after the config IP address is enabled.

Step 1 Log in to the management console.

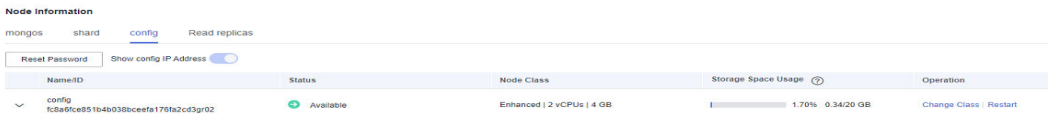
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name. The **Basic Information** page is displayed.

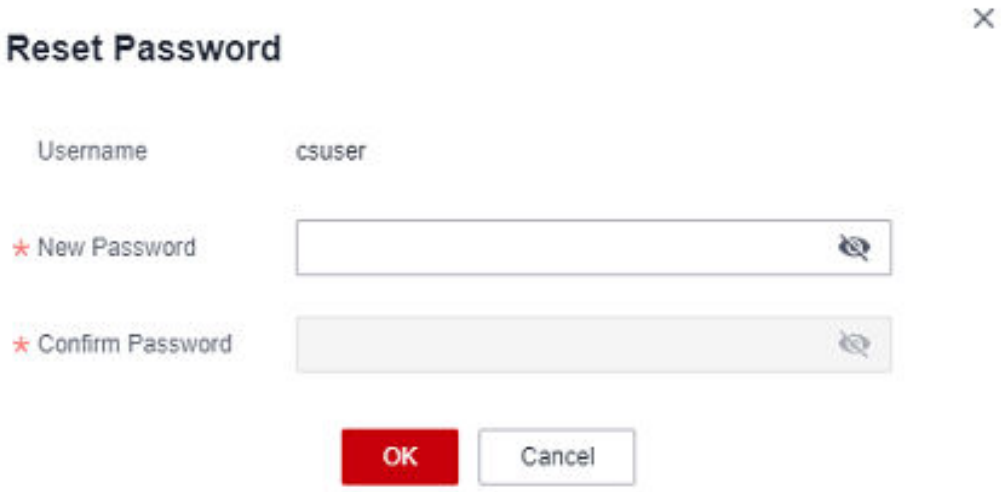
Step 5 In the **Node Information** area, click the **config** tab.

Figure 2-13 config nodes



Step 6 Click **Reset Password**.

Figure 2-14 Resetting a password



Step 7 Enter the new password and click **OK**.

----End

Follow-up Operations

After the connection addresses of the shard or config nodes are enabled, you can connect to the shard or config nodes using Mongo Shell. The procedure is similar to that for connecting to a dds mongos node. For details, see **Connecting to a Cluster Instance Using Mongo Shell**.

2.3 Changing a Private IP Address

Scenarios

After data is migrated from an on-premises database or other cloud databases to DDS, the private IP address of the database may be changed. DDS allows you to change the private IP address, simplifying and accelerating the migration process.

Precautions

Changing the private IP address of a node will invalidate the previous private IP address. If an EIP is bound to the node, do not unbind the EIP during the change of the private IP address. After the change, the new private IP address is bound to the EIP.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

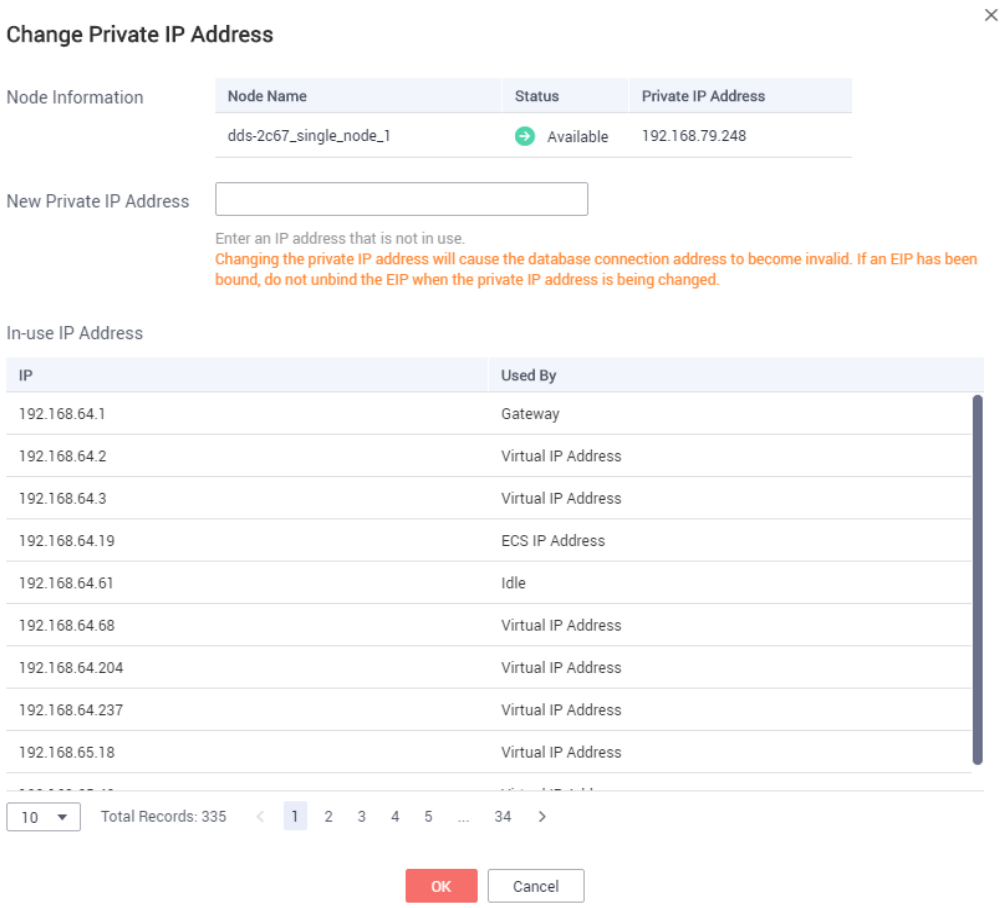
Step 4 On the **Instances** page, click the instance name. The **Basic Information** page is displayed.

Alternatively, you can click **Connections** in the navigation pane on the left to go to the **Basic Information** page.

Step 5 In the **Node Information** area, locate the target node and click **Change Private IP Address** in the **Operation** column.

Step 6 In the displayed dialog box, enter a private IP address that is not in use and click **OK**.

Figure 2-15 Changing a private IP address



Step 7 In the **Node Information** area, locate the target node and view the new private IP address.

----End

2.4 Changing a Database Port

Scenarios

This section describes how to change a database port.

Precautions

- For security purposes, the database port cannot be modified when the instance is in any of the following statuses:
  - Frozen
  - Restarting
  - Adding node
  - Switching SSL
  - Changing instance class

  - Deleting node
  - The storage space is being expanded.
  - Abnormal
- The default port of a DB instance is 8635. After a DB instance is created, you can change its port number to a value ranging from 2100 to 65535 (excluding 12017 and 33071).

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the **Network Information** area on the **Basic Information** page, click in the **Database Port** field to change the database port.

In the navigation pane on the left, choose **Connections** and click in the **Database Port** field in the **Basic Information** area to change the database port.

NOTE

The database port ranges from 2100 to 65535 (excluding 12017 and 33071).

- To submit the change, click . This process takes about 1 to 5 minutes.
- To cancel the change, click .

Step 6 View the modification result.

----End

2.5 Applying for and Modifying a Private Domain Name

Scenarios

You can apply for a private domain name and connect to DDS instances using the private domain name.

Precautions

- After a private domain name is generated, changing the private IP address will interrupt database connections. Exercise caution when performing this operation.
- You need to apply for the permissions needed to use private domain names. You can choose **Service Tickets** > **Create Service Ticket** to apply for the permissions.

- When this function is enabled, you need to apply for a domain name for an existing instance. A domain name is automatically applied for a new instance.
- This function is available in the following regions: CN North-Beijing1, CN North-Beijing4, CN North-Ulanqab1, CN East-Shanghai1, CN East-Shanghai2, CN South-Guangzhou, CN Southwest-Guiyang1, AP-Singapore, AP-Bangkok, CN-Hong Kong, LA-Mexico City2, LA-Sao Paulo1, and ME-Riyadh.

Applying for a Private Domain Name

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

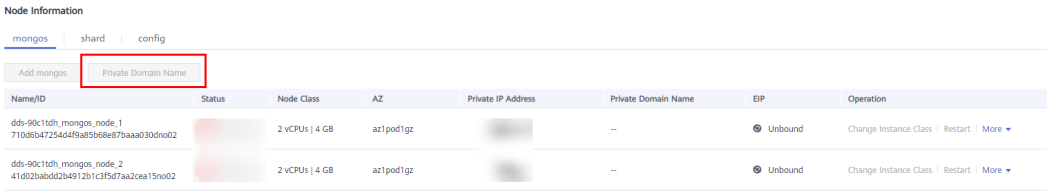
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name and go to the **Basic Information** page.

Step 5 In the **Node Information** area on the **Basic Information** page, click **Private Domain Name**.

Alternatively, in the navigation pane on the left, choose **Connections**. In the **Basic Information** area on the **Private Connection** tab, click **Private Domain Name**.

Figure 2-16 Applying for a private domain name



Step 6 In the **Node Information** area on the **Basic Information** page, view the generated private domain names in the **Private Domain Name** column.

Alternatively, click **Connections** in the navigation pane on the left. In the **Basic Information** area on the displayed page, view the generated private domain names in the **Private Domain Name** column.

----End

Modifying a Private Domain Name

You can change the private domain name of an existing DB instance.

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the **Node Information** area on the **Basic Information** page, choose **More** > **Change Private Domain Name** in the **Operation** column.

Alternatively, choose **Connections** in the navigation pane on the left. In the lower part of the **Basic Information** area on the **Private Connection** tab, choose **More** > **Change Private Domain Name** in the **Operation** column.

Step 6 In the displayed dialog box, enter a new private domain name. Click **OK**. After the private domain name is changed, it takes about 5 minutes for the change to take effect.

NOTE

- Only the prefix of a private domain name can be modified.
- The prefix of a private domain name can contain 8 to 56 characters, and can include only letters and digits.
- The new private domain name must be different from the existing ones.

Step 7 If you have enabled the operation protection function, click **Send Code** in the displayed **Identity Verification** dialog box and enter the obtained verification code. Then, click **OK**.

Two-factor authentication improves the security of your account and cloud product. For details about how to enable operation protection, see .

----End

3 Data Migration

3.1 Migration Scheme Overview

DDS provides multiple migration schemes to migrate MongoDB databases in different service scenarios.

Table 3-1 Migration schemes

Scenario	Migration Types	References
Migrating data using the export and import tools	Full	• **Migrating Data Using mongoexport and mongoimport** • **Migrating Data Using mongodump and mongorestore**
Migrating data from other cloud MongoDB to DDS	Full +incremental	**Migrating from Other Cloud MongoDB to DDS**
Migrating data from on-premises MongoDB to DDS	Full +incremental	**Migrating from On-Premises MongoDB to DDS**
Migrating data from ECS-hosted MongoDB to DDS	Full +incremental	**Migrating from ECS MongoDB Databases to DDS**
Migrating data from DDS to MongoDB	Full +incremental	**Migrating from DDS to MongoDB**

3.2 Migrating Data Using DRS

Data Replication Service (DRS) helps migrate your databases to DDS DB instances. During the migration, the source remains operational even if a transfer is interrupted, thereby minimizing application downtime.

Prerequisites

To improve the stability and security of your migration ensure that your instances meet the migration requirements described in Migration Preparations.

Migration Types

- **Full migration**

  This migration type is suitable for scenarios where some service interruptions are acceptable. All objects and data in non-system databases are migrated to the destination database in a single batch. The objects include tables, views, and stored procedures. If you perform a full migration, stop operations on the source database, or data generated in the source database during the migration will result in inconsistencies with the destination database.

- **Full+Incremental migration**

  This migration type allows you to migrate data without interrupting services. After a full migration initializes the destination database, an incremental migration initiates and parses logs to ensure data consistency between the source and destination databases. If you select the **Full+Incremental** migration type, data generated during the full migration will be synchronized to the destination database with zero downtime, ensuring that both the source and destination databases remain accessible throughout the process.

Supported Source and Destination Databases

Table 3-2 Supported databases

Source DB	Destination DB
• On-premises MongoDB (versions 3.2, 3.4, 3.6, 4.0, 4.2, 4.4, and 5.0) • ECS-hosted MongoDB (versions 3.2, 3.4, 3.6, 4.0, 4.2, 4.4, and 5.0) • Other cloud MongoDB (versions 3.2, 3.4, 3.6, 4.0, 4.2, 4.4, and 5.0) • DDS DB instances (versions 3.2, 3.4, 4.0, 4.2, 4.4, and 5.0) **NOTE** If the source database is a DDS 3.2 cluster instance, only full migration is supported. DDS 5.0 supports replica sets only. 1. If the source database is a DDS DB instance, the source DB engine is DDS. Otherwise, the source DB engine is MongoDB.	• DDS DB instances (versions 3.4, 4.0, 4.2, 4.4, and 5.0) **NOTE** The destination database version must be the same as or later than the source database version. DDS 5.0 supports replica sets only.

Supported Migration Objects

Different types of migration tasks support different migration objects. For details, see **Table 3-3**. DRS will automatically check the objects you selected before the migration.

Table 3-3 Migration objects

Type	Precautions
Migration objects	• Object level: table level, database level, or instance level (full migration). • Supported migration objects: – Associated objects must be migrated at the same time to avoid migration failure caused by missing associated objects. Common associations: collections referenced by views, and views referenced by views – Replica set: Only collections (including validator and capped collections), indexes, and views can be migrated. – Cluster: Only collections (including validator and capped collections), shard keys, indexes, and views can be migrated. – Single node: Only collections (including validator and capped collections), indexes, and views can be migrated. – Only user data and source database account information can be migrated. The system databases (for example, local, admin, and config) and system collection cannot be migrated. If service data is stored in a system database, run the **renameCollection** command to move the service data to the user database. – The statement for creating a view cannot contain a regular expression. – Collections that contain the **_id** field without indexes are not supported. – The first parameter of **BinData()** cannot be **2**. – If ranged sharding is used, **maxKey** cannot be used as the primary key. **NOTE** The objects that can be migrated have the following constraints: • The source database name cannot contain /\."$ or spaces. The collection name and view name cannot start with **system.** or contain the dollar sign ($).

Database Account Permission Requirements

To start a migration task, the source and destination database users must have permissions listed in the following table. Different types of migration tasks require different permissions. For details, see **Table 3-4**. DRS automatically checks the database account permissions in the pre-check phase and provides handling suggestions.

NOTE

- You are advised to create an independent database account for DRS task connection to prevent task failures caused by database account password modification.

Table 3-4 Database account permission

Type	Full migration	Full+Incremental Migration
Source database user	• Replica set: The source database user must have the readAnyDatabase permission for the admin database. • Single node: The source database user must have the readAnyDatabase permission for the admin database. • Cluster: The source database user must have the readAnyDatabase permission for the admin database and the read permission for the config database. • To migrate accounts and roles of the source database, the source and destination database users must have the read permission for the **system.users** and **system.roles** system tables of the admin database.	• Replica set: The source database user must have the readAnyDatabase permission for the admin database and the read permission for the local database. • Single node: The source database user must have the readAnyDatabase permission for the admin database and the read permission for the local database. • Cluster: The source dds mongos node user must have the readAnyDatabase permission for the admin database and the read permission for the config database. The source shard node user must have the readAnyDatabase permission for the admin database and the read permission for the local database. • To migrate accounts and roles of the source database, the source and destination database users must have the read permission for the **system.users** and **system.roles** system tables of the admin database.
Destination database user	The destination database user must have the dbAdminAnyDatabase permission for the admin database and the readWrite permission for the destination database. If the destination database is a cluster instance, the database user must have the clusterManager permission for the admin database.	

NOTE

For example, the source database user must have the readAnyDatabase permission for the admin database and the read permission for the config database.

```
db.grantRolesToUser("Username",[{role:"readAnyDatabase",db:"admin"},
{role:"read",db:"config"}])
```

Migration Operations

For details, see MongoDB Database Migration in *Data Replication Service Best Practices*.

3.3 Migrating Data Using mongoexport and mongoimport

Scenarios

mongoexport and mongoimport are backup and restoration tools provided by the MongoDB client. You can install a MongoDB client on the local device or ECS and use the mongoexport and mongoimport tools to migrate your on-premises MongoDB databases or other cloud MongoDB databases to DDS instances.

Before migrating data from a MongoDB database to DDS, transfer data to a .json file using the mongoexport tool. This section describes how to import the data from the .json file to DDS using the mongoimport tool on the ECS or the device that can access DDS.

Precautions

- The mongoexport and mongoimport tools support only full migration. To ensure data consistency, stop services on the source database and stop writing data to the source database before the migration.
- You are advised to perform the migration during off-peak hours to avoid impacting services.
- The admin and local system databases cannot be migrated.
- Make sure that no service set has been created in the system databases admin and local in the source database. If there is already a service set, migrate them out of the system databases admin and local before migration.
- Before importing data, ensure that the necessary indexes are there on the source database. Delete any unnecessary indexes and create any necessary indexes before migration.
- If you choose to migrate a sharded cluster, you must create a set of shards in the destination database and configure sharding. In addition, indexes must be created before migration.

Prerequisites

1. An ECS or a device that can access DDS is ready for use.
   - To bind an EIP to a DB instance:

i. Bind an EIP to a node in the instance. For details about how to bind an EIP to a node, see "Binding an EIP" in *Getting Started with Document Database Service*.

ii. Ensure that your local device can access the EIP that has been bound to the DB instance.

2. A migration tool has been installed on the prepared ECS.

   For details about how to install a migration tool, see **MongoDB Command Line Database Tools Download**

Exporting Data

Step 1 Log in to the ECS or the device that can access DDS.

Step 2 Use the mongoexport tool to transfer data from the source database to a .json file.

The SSL connection is used as an example. If you select a common connection, delete **--ssl --sslAllowInvalidCertificates** from the following command.

./**mongoexport --host** *<DB_ADDRESS>* **--port** *<DB_PORT>* **--ssl --sslAllowInvalidCertificates --type json --authenticationDatabase** *<AUTH_DB>* **-u** *<DB_USER>* **--db** *<DB_NAME>* **--collection** *<DB_COLLECTION>* **--out** *<DB_PATH>*

- **DB_ADDRESS** is the database address.
- **DB_PORT** is the database port.
- **AUTH_DB** is the database for storing DB_USER information. Generally, this value is **admin**.
- **DB_USER** is the database user.
- **DB_NAME** is the name of the database from which data will be exported.
- **DB_COLLECTION** is the collection of the database from which data will be exported.
- **DB_PATH** is the path where the .json file is located.

Enter the database administrator password when prompted:

```
Enter password:
```

The following is an example. After the command is executed, the **exportfile.json** file will be generated:

./mongoexport --host 192.168.1.21 --port 8635 --ssl --sslAllowInvalidCertificates --type json --authenticationDatabase admin -u rwuser --db test02 --collection Test --out /tmp/mongodb/export/exportfile.json

Step 3 View the results.

If information similar to the following is displayed, the data has been successfully exported. **x** is the number of exported data records.

```
exported x records
```

Step 4 Compress the exported .json file.

gzip exportfile.json

Compressing the file helps reduce the time needed to transmit the data. The compressed file is **exportfile.json.gz**.

----End

Importing Data

Step 1 Log in to the ECS or whichever device you will be using to access DDS.

Step 2 Upload the data to be imported to the ECS or the device.

Select an uploading method based on the OS you are using.

- In Linux, for example, you can use secure copy protocol (SCP):

  scp *<IDENTITY_FILE>*
  <REMOTE_USER>@*<REMOTE_ADDRESS>*:*<REMOTE_DIR>*

  - **IDENTITY_FILE** is the directory where the **exportfile.json.gz** file is located. The file access permission is 600.
  - **REMOTE_USER** is the ECS OS user.
  - **REMOTE_ADDRESS** is the ECS address.
  - **REMOTE_DIR** is the directory of the ECS to which the **exportfile.json.gz** file is uploaded.
- In Windows, upload **exportfile.json.gz** to the ECS using file transfer tools.

Step 3 Decompress the package.

gzip -d *exportfile.json.gz*

Step 4 Import the JSON file to the DDS database.

The SSL connection is used as an example. If you select a common connection, delete **--ssl --sslAllowInvalidCertificates** from the following command.

./**mongoimport --host** <*DB_ADDRESS*> **--port** <*DB_PORT*> **--ssl --sslAllowInvalidCertificates --type json --authenticationDatabase** <*AUTH_DB*> **-u** <*DB_USER*> **--db** <*DB_NAME*> **--collection** <*DB_COLLECTION*> **--file** <*DB_PATH*>

- **DB_ADDRESS** indicates the DB instance IP address.
- **DB_PORT** indicates the database port.
- **AUTH_DB** indicates the database that authenticates DB_USER. Generally, this value is **admin**.
- **DB_USER** indicates the account name of the database administrator.
- **DB_NAME** indicates the name of the database to which data will be imported.
- **DB_COLLECTION** indicates the collection of the database to which data will be imported.
- **DB_PATH** indicates the path where the .json file is located.

Enter the database administrator password when prompted:

```
Enter password:
```

The following is an example:

./mongoimport --host 192.168.1.21 --port 8635 --ssl --sslAllowInvalidCertificates --type json --authenticationDatabase admin -u rwuser --db test02 --collection Test --file /tmp/mongodb/export/exportfile.json

Step 5 View the results.

If information similar to the following is displayed, the data has been successfully imported. **x** is the number of imported data records.

```
imported x records
```

----End

3.4 Migrating Data Using mongodump and mongorestore

Scenarios

mongodump and mongorestore are backup and restoration tools provided by the MongoDB client. You can install a MongoDB client on the local device or ECS and use the mongodump and mongorestore tools to migrate your on-premises MongoDB databases or other cloud MongoDB databases to DDS instances.

Precautions

- The mongodump and mongorestore tools support only full migration. To ensure data consistency, stop services on the source database and stop writing data to the source database before the migration.
- You are advised to perform the migration during off-peak hours to avoid impacting services.
- The admin and local system databases cannot be migrated.
- The file exported by mongodump is a BSON binary file. The mongorestore uses this binary backup file to restore data to a DB instance.
- Make sure that no service set has been created in the system databases admin and local in the source database. If there is already a service set, migrate them out of the system databases admin and local before migration.
- Before importing data, ensure that the necessary indexes are there on the source database. Delete any unnecessary indexes and create any necessary indexes before migration.
- If you choose to migrate a sharded cluster, you must create a set of shards in the destination database and configure sharding. In addition, indexes must be created before migration.
- If the backup using the mongodump tool fails (for example, an error is reported when the backup progress reaches 97%), you are advised to increase the storage space of the VM that fails to be backed up and reserve some redundant space before performing the backup again.
- User **rwuser** can only operate service database tables. You are advised to specify databases and tables to import and export only service data. Otherwise, the insufficient permission problem may occur during full import and export.

- For details about how to restore backup data to an on-premises database, see **Restoring Data to an On-Premises Database**.

Prerequisites

1. Prepare an ECS or a device that can access DDS.
   - To bind an EIP to a DB instance:
     i. Bind an EIP to a node in the DB instance. For details about how to bind an EIP to a node, see "Binding an EIP" in the *Getting Started with Document Database Service*.
     ii. Ensure that your local device can access the EIP that has been bound to the DB instance.
2. A migration tool has been installed on the prepared ECS.

   For details about how to install a migration tool, see **MongoDB Command Line Database Tools Download**

   NOTE

   - The MongoDB client version must match the instance version. Otherwise, compatibility issues may occur.

Exporting Data

Step 1 Log in to the ECS or the device that can access DDS.

Step 2 Back up the source database data using the mongodump tool.

An SSL connection is used in this example. If you select an unencrypted connection, delete **--ssl --sslCAFile** *<FILE_PATH>* **--sslAllowInvalidCertificates** from the following command.

./mongodump --host *<DB_HOST>* **--port** *<DB_PORT>* **--authenticationDatabase** *<AUTH_DB>* **-u <DB_USER> --ssl --sslCAFile** *<FILE_PATH>* **--sslAllowInvalidCertificates --db** *<DB_NAME>* **--collection** *<DB_COLLECTION>* **--gzip --archive=***<Name of the backup file that contains the file path>*

Table 3-5 Parameter description

Parameter	Description
<DB_HOST>	Database address
<DB_PORT>	Database port
<DB_USER>	Database username
<AUTH_DB>	Database that stores *<DB_USER>* information. Generally, the value is **admin**.
<FILE_PATH>	Path for storing the root certificate
<DB_NAME>	The name of the database to be migrated.
<DB_COLLECTION>	Collection in the database to be migrated

Enter the database administrator password when prompted:

```
Enter password:
```

After the command is executed, the file specified by **archive** is the final backup file. The following command uses **backup.tar.gz** as an example.

./mongodump --host 192.168.*xx.xx* --port 8635 --authenticationDatabase admin -u rwuser --ssl --sslCAFile /tmp/ca.crt --sslAllowInvalidCertificates --db test --collection usertable --gzip --archive=backup.tar.gz

```
2019-03-04T18:42:10.687+0800    writing admin.system.users to
2019-03-04T18:42:10.688+0800    done dumping admin.system.users (1 document)
2019-03-04T18:42:10.688+0800    writing admin.system.roles to
2019-03-04T18:42:10.690+0800    done dumping admin.system.roles (0 documents)
2019-03-04T18:42:10.690+0800    writing admin.system.version to
2019-03-04T18:42:10.691+0800    done dumping admin.system.version (2 documents)
2019-03-04T18:42:10.691+0800    writing test.test_collection to
2019-03-04T18:42:10.691+0800    writing admin.system.profile to
2019-03-04T18:42:10.692+0800    done dumping admin.system.profile (4 documents)
2019-03-04T18:42:10.695+0800    done dumping test.test_collection (198 documents)
```

----End

Importing Data

Step 1 Log in to the ECS or whichever device you will be using to access DDS.

Step 2 Upload the data to be imported to the ECS or the device.

Select an uploading method based on the OS you are using.

- In Linux, for example, you can use secure copy protocol (SCP):

  scp -r *<IDENTITY_DIR>* *<REMOTE_USER>*@*<REMOTE_ADDRESS>*:*<REMOTE_DIR>*

  Table 3-6 Parameter description

Parameter	Description
<IDENTITY_DIR>	Directory where the backup folder is located.
<REMOTE_USER>	User of ECS OS in **Step 1**
<REMOTE_ADDRESS>	IP address of the ECS in **Step 1**
<REMOTE_DIR>	Directory of the ECS to be imported

- In Windows, upload the backup directory to the ECS using a file transfer tool.

Step 3 Import the backup data to DDS.

An SSL connection is used in this example. If you use an unencrypted connection, delete **--ssl --sslCAFile** <*FILE_PATH*> **--sslAllowInvalidCertificates** from the following command.

./mongorestore --host *<DB_HOST>* **--port** *<DB_PORT>* **--authenticationDatabase** *<AUTH_DB>* **-u***<DB_USER>* **--ssl --sslCAFile** <*FILE_PATH*> **--sslAllowInvalidCertificates --db** <*DB_NAME*> **--collection** <*DB_COLLECTION*> **--gzip --archive=**<*Name of the backup file that contains the file path*>

Table 3-7 Parameter description

Parameter	Description
<DB_HOST>	DDS database address
<DB_PORT>	Database port
<AUTH_DB>	The database that authenticates *DB_USER*. Generally, the value is **admin**.
<DB_USER>	Account name of the database administrator. The default value is **rwuser**.
<*FILE_PATH*>	Path for storing the root certificate
<*DB_NAME*>	The name of the database to be migrated.
<*DB_COLLECTION*>	Collection in the database to be migrated

Enter the database administrator password when prompted:

```
Enter password:
```

The following is an example:

./mongorestore --host 192.168.*xx.xx* **--port 8635 --authenticationDatabase admin -u rwuser --ssl --sslCAFile /tmp/ca.crt --sslAllowInvalidCertificates --db test --collection usertable --gzip --archive=backup.tar.gz**

```
2019-03-05T14:19:43.240+0800    preparing collections to restore from
2019-03-05T14:19:43.243+0800    reading metadata for test.test_collection from dump/test/
test_collection.metadata.json
2019-03-05T14:19:43.263+0800    restoring test.test_collection from dump/test/test_collection.bson
2019-03-05T14:19:43.271+0800    restoring indexes for collection test.test_collection from metadata
2019-03-05T14:19:43.273+0800    finished restoring test.test_collection (198 documents)
2019-03-05T14:19:43.273+0800    restoring users from dump/admin/system.users.bson
2019-03-05T14:19:43.305+0800    roles file 'dump/admin/system.roles.bson' is empty; skipping roles
restoration
2019-03-05T14:19:43.305+0800    restoring roles from dump/admin/system.roles.bson
2019-03-05T14:19:43.333+0800    done
```

----End

Related Issues

When you back up the entire instance using mongodump and mongorestore, the permission verification fails.

- Cause

The **rwuser** user has limited permissions on the **admin** and **config** databases of the instance. As a result, the permission verification fails.

- Solution

  Grant permissions on certain databases and tables to the user.

4 Performance Tuning

4.1 Parameters

Scenarios

Parameters are key configuration items in a database system. Improper parameter settings may adversely affect database performance. This section describes some important parameters. For details on parameter descriptions, visit **MongoDB official website**.

For details about how to change parameter values on the console, see **Modifying DDS DB Instance Parameters**.

Parameter Description

- **enableMajorityReadConcern**

  This parameter indicates whether data read has been acknowledged by a majority of nodes.

  The default value is **false**, indicating that data read is returned after being acknowledged by a single node.

  If this parameter is set to **true**, data read is returned after being acknowledged by a majority of nodes. This operation will increase the size of the LAS file, resulting in high CPU usage and disk usage.

  In DDS, read concern cannot be set to majority. If majority read concern is required, you can set write concern to majority, indicating that data is written to a majority of nodes. In this way, data on most nodes is consistent. Then, by reading data from a single node, it can be ensured that the data has been written to a majority of nodes, and there are no dirty reads.

  NOTE

  Write concern and read concern respectively specify the write and read policies for MongoDB.

  If read concern is set to majority, data read by users has been written to a majority of nodes and will not be rolled back to avoid dirty reads.

- **failIndexKeyTooLong**

The default value is **true**.

This parameter cannot be modified to avoid an excessively long index key.

- **net.maxIncomingConnections**

  This parameter indicates the maximum number of concurrent connections that dds mongos or mongod can accept. The default value depends on the **instance specifications**. This parameter is displayed as **default** before being set, indicating that the parameter value varies with the memory specifications.

- **security.javascriptEnabled**

  The default value is **false**.

  This parameter indicates whether JavaScript scripts can be executed on mongod. For security purposes, the default value is **false**, indicating that JavaScript scripts cannot be executed on mongod, and the **mapreduce** and **group** commands cannot be used.

- **disableJavaScriptJIT**

  The default value is **true**.

  This parameter indicates whether to disable JavaScript JIT compilation. JavaScript JIT compilation enables just-in-time (JIT) compilation to improve the performance of running scripts.

  disableJavaScriptJIT: The default value is **true**, indicating that JavaScript JIT compilation is disabled. To enable JavaScript JIT compilation, set **disableJavaScriptJIT** to **false**.

- **operationProfiling.mode**

  The parameter value is **slowOp** by default.

  This parameter indicates the level of the database analyzer.

  This parameter supports the following values:

  – The default value is **slowOp**, indicating that the collector records statements whose response time exceeds the threshold.

  – The value **off** indicates that the analyzer is disabled and does not collect any data.

  – The value **all** indicates that the collector collects data of all operations.

- **operationProfiling.slowOpThresholdMs**

  The default value is **500** and the unit is ms.

  This parameter indicates the threshold for slow queries in the unit of ms. Queries that take longer than the threshold are deemed as slow queries.

  Unless otherwise specified, setting the value to 500 ms is recommended.

- **maxTransactionLockRequestTimeoutMillis**

  The value ranges from **5** to **100**, in milliseconds. The default value is **5**.

  This parameter specifies the time for a transaction to wait for locks. If the time is exceeded, the transaction is rolled back.

4.2 Read and Write Performance

Common Check Items

1. If the error message Timeout is displayed in the database, check whether the number of connections to the instance reaches the upper limit.
   - Check method: View the **monitoring metric** to check whether the **maximum number of active connections** has been reached.
   - Solution: See **What Can I Do If the Number of Connections of an Instance Reaches Its Maximum?**
2. Check whether the instance is properly connected.
   - Check method: Check whether multiple dds mongos nodes in a cluster instance are connected and whether both the primary and secondary nodes in a replica set instance are connected.
   - Solution: If you connect to a cluster instance, connect to multiple dds mongos nodes at the same time to share the load and improve availability. If you connect to a replica set instance, connect to both the primary and secondary nodes. This improves read/write performance and prevents errors from being reported when data is written from the client after a primary/secondary switchover.
3. Check whether the monitoring metrics of the instance are normal.
   - Check method: View **monitoring metrics** to check the CPU usage and memory usage.
   - Solution: If the CPU and memory metrics are abnormal, check whether the client service load is too centralized or instance data is too intensive. If the client service load is too centralized, optimize the client architecture. If data is too intensive, shard data.
4. Check whether there are too many slow query logs.

   Check method: For details, see **Viewing Slow Query Logs**.

   Solution: For details, see **Slow Request Locating**.

Other Precautions

- During the query, select only the fields that need to be returned. When modifying data, modify only the fields that need to be modified. Do not directly store all modifications of the entire object. In this way, the network and processing loads are reduced.
- In the same service scenario, reduce the number of interactions with the database and query data at a time if possible.
- In a DB instance, the total number of databases cannot exceed 200, and the total number of collections cannot exceed 500.
- Before bringing a service online, perform a load test to measure the performance of the database in peak hours.
- Do not execute a large number of concurrent transactions at the same time or leave a transaction uncommitted for a long time.
- Before bringing a service online, execute the query plan to check the query performance for all query types.

- Check the performance baseline of the instance specifications and analyze whether the current service requirements reach the upper limit.

4.3 Sharding

You can shard a large-size collection for a sharded cluster instance. Sharding distributes data across different machines to make full use of the storage space and compute capability of each shard.

CAUTION

If sharding is enabled for a non-empty collection, the collection is locked and services are blocked. The time required for sharding increases with the amount of data in the collection. To minimize the impact, before sharding, ensure that the target table's workload is running during off-peak hours.

Number of Shards

The following is an example using database **mytable**, collection **mycoll**, and the field **name** as the shard key.

Step 1 Log in to a sharded cluster instance using Mongo Shell.

Step 2 Check whether a collection has been sharded.

```
use <database>
db.<collection>.getShardDistribution()
```

Example:

```
use mytable
db.mycoll.getShardDistribution()
```

```
mongos> db.mycoll.getShardDistribution()
Collection test.mycoll is not sharded.
```

Step 3 Enable sharding for the databases that belong to the cluster instance.

- Method 1

  ```
  sh.enableSharding("<database>")
  ```

  Example:

  ```
  sh.enableSharding("mytable")
  ```

- Method 2

  ```
  use admin
  db.runCommand({enablesharding:"<database>"})
  ```

Step 4 Shard a collection.

- Method 1

  ```
  sh.shardCollection("<database>.<collection>",{"<keyname>":<value> })
  ```

  Example:

  ```
  sh.shardCollection("mytable.mycoll",{"name":"hashed"},false,{numInitialChunks:5})
  ```

- Method 2


```
use admin
db.runCommand({shardcollection:"<database>.<collection>",key:{"keyname":<value> }})
```

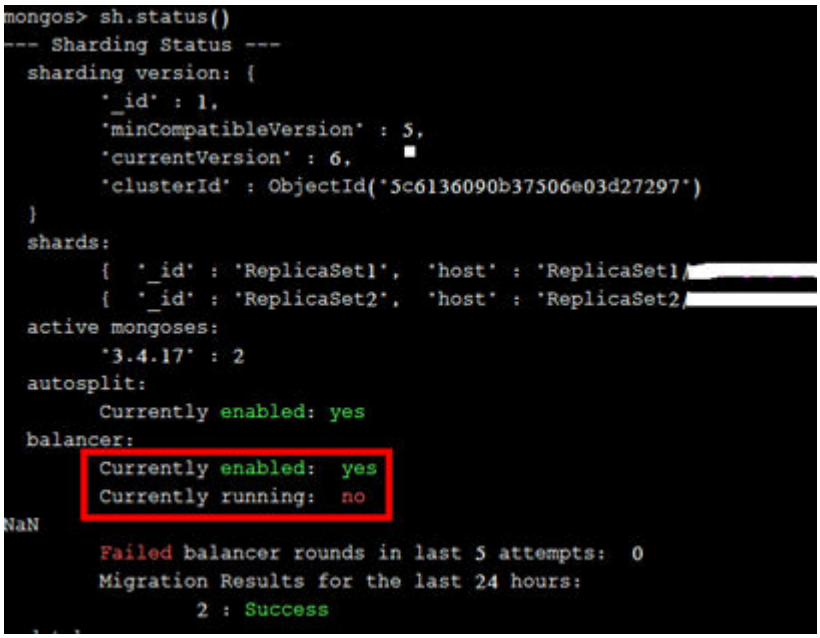
Table 4-1 Parameter description

Parameter	Description
<database>	Database name
<collection>	Collection name.
<keyname>	Shard key. Cluster instances are sharded based on the value of this parameter. Select a proper shard key for the collection based on your service requirements. For details, see **Selecting a Shard Key**.
<value>	The sort order based on the range of the shard key. • 1: Ascending indexes • -1: Descending indexes • hashed: indicates that hash sharding is used. Hashed sharding provides more even data distribution across the sharded cluster. For details, see **sh.shardCollection()**.
numInitialChunks	Optional. The minimum number of shards initially created is specified when an empty collection is sharded using a hashed shard key.

Step 5 Check the data storage status of the database on each shard.

```
sh.status()
```

Example:



----End

Selecting a Shard Key

- **Background**

  Each sharded cluster contains collections as its basic unit. Data in the collection is partitioned by the shard key. Shard key is a field in the collection. It distributes data evenly across shards. If you do not select a proper shard key, the cluster performance may deteriorate, and the sharding statement execution process may be blocked.

  Once the shard key is determined it cannot be changed. If no shard key is suitable for sharding, you need to use a sharding policy and migrate data to a new collection for sharding.

- **Characteristics of proper shard keys**
  - All inserts, updates, and deletes are evenly distributed to all shards in a cluster.
  - The distribution of keys is sufficient.
  - Rare scatter-gather queries.

  If the selected shard key does not have all the preceding features, the read and write scalability of the cluster is affected. For example, if the workload of the find() operation is unevenly distributed in the shards, hot shards will be generated. Similarly, if your write load (inserts, updates, and deletes) is not uniformly distributed across your shards, then you could end up with a hot shard. Therefore, you need to adjust the shard keys based on service requirements, such as read/write status, frequently queried data, and written data.

  After existing data is sharded, if the **filter** field of the update request does not contain shard keys and **upsert:true** or **multi:false**, the update request will report an error and return message "An upsert on a sharded collection must contain the shard key and have the simple collation.".

- **Judgment criteria**

You can use the dimensions provided in **Table 4-2** to determine whether the selected shard keys meet your service requirements:

Table 4-2 Reasonable shard keys

Identification Criteria	Description
Cardinality	Cardinality refers to the capability of dividing chunks. For example, if you need to record the student information of a school and use the age as a shard key, data of students of the same age will be stored in only one data segment, which may affect the performance and manageability of your clusters. A much better shard key would be the student number because it is unique. If the student number is used as a shard key, the relatively large cardinality can ensure the even distribution of data.
Write distribution	If a large number of write operations are performed in the same period of time, you want your write load to be evenly distributed over the shards in the cluster. If the data distribution policy is ranged sharding, a monotonically increasing shard key will guarantee that all inserts go into a single shard.
Read distribution	Similarly, if a large number of read operations are performed in the same period, you want your read load to be evenly distributed over the shards in a cluster to fully utilize the computing performance of each shard.
Targeted read	The dds mongos query router can perform either a targeted query (query only one shard) or a scatter/gather query (query all of the shards). The only way for the dds mongos to be able to target a single shard is to have the shard key present in the query. Therefore, you need to pick a shard key that will be available for use in the common queries while the application is running. If you pick a synthetic shard key, and your application cannot use it during typical queries, all of your queries will become scatter/gather, thus limiting your ability to scale read load.

Choosing a Distribution Policy

A sharded cluster can store a collection's data on multiple shards. You can distribute data based on the shard keys of documents in the collection.

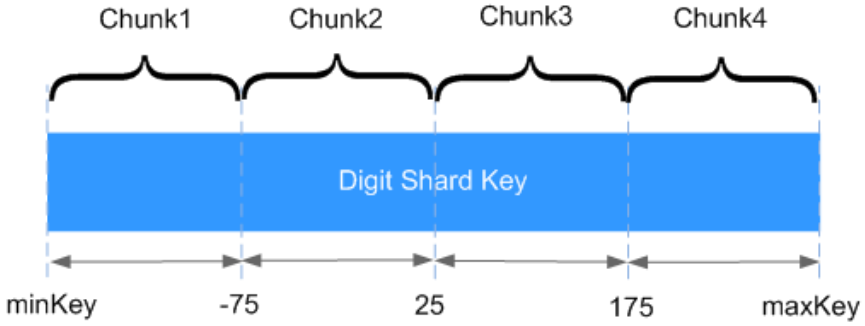
There are two data distribution policies: ranged sharding and hashed sharding. For details, see **Step 4**.

The following describes the advantages and disadvantages of the two methods.

- **Ranged sharding**

Ranged-based sharding involves dividing data into contiguous ranges determined by the shard key values. If you assume that a shard key is a line stretched out from positive infinity and negative infinity, each value of the shard key is the mark on the line. You can also assume small and separate segments of a line and that each chunk contains data of a shard key within a certain range.

Figure 4-1 Distribution of data



As shown in the preceding figure, field **x** indicates the shard key of ranged sharding. The value range is [*minKey*,*maxKey*] and the value is an integer. The value range can be divided into multiple chunks, and each chunk (usually 64 MB) contains a small segment of data. For example, chunk 1 contains all documents in range [minKey, -75] and all data of each chunk is stored on the same shard. That means each shard containing multiple chunks. In addition, the data of each shard is stored on the config server and is evenly distributed by dds mongos based on the workload of each shard.

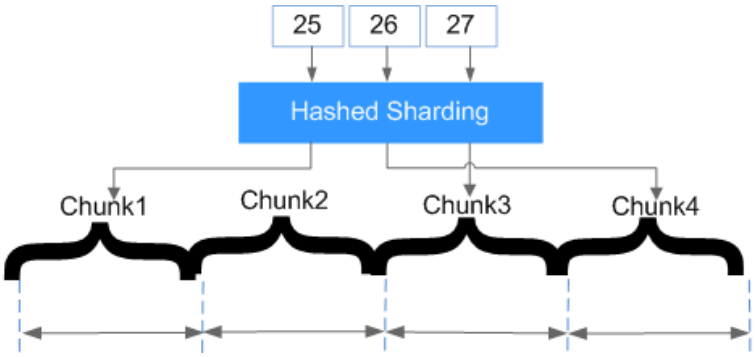
Ranged sharding can easily meet the requirements of query in a certain range. For example, if you need to query documents whose shard key is in range [-60,20], dds mongos only needs to forward the request to chunk 2.

However, if shard keys are in ascending or descending order, newly inserted documents are likely to be distributed to the same chunk, affecting the expansion of write capability. For example, if **_id** is used as a shard key, the high bits of **_id** automatically generated in the cluster are ascending.

- **Hashed sharding**

  Hashed sharding computes the hash value (64-bit integer) of a single field as the index value; this value is used as your shard key to partition data across your shared cluster. Hashed sharding provides more even data distribution across the sharded cluster because documents with similar shard keys may not be stored in the same chunk.

Figure 4-2 Distribution of data



Hashed sharding randomly distributes documents to each chunk, which fully expands the write capability and makes up for the deficiency of ranged sharding. However, queries in a certain range need to be distributed to all backend shards to obtain documents that meet conditions, resulting in low query efficiency.

4.4 High CPU Usage

Scenarios

If your CPU usage reaches 80%, a CPU bottleneck exists. In this case, data reads and writes are slow, affecting your services.

The following describes how to analyze current slow queries. After the analysis and optimization, query performance will be improved and indexes will be used more efficiently.

Analyzing Current Queries

1. Connect to an instance using Mongo Shell.

   To enable public access, see:

   - **Connecting to a Cluster Instance over a Public Network**
   - **Connecting to a Replica Set Instance over a Public Network**
   - **Connecting to a Single Node Instance over a Public Network**

   To access an instance over a private network, see:

   - **Connecting to a Cluster Instance over a Private Network**
   - **Connecting to a Replica Set Instance over a Private Network**
   - **Connecting to a Single Node Instance over a Private Network**

2. Run the following command to view the operations being performed on the database:

   db.currentOp()

   Command output:

   ```
   {
         "raw" : {
               "shard0001" : {
                     "inprog" : [
   ```



```
                {
                        "desc" : "StatisticsCollector",
                        "threadId" : "140323686905600",
                        "active" : true,
                        "opid" : 9037713,
                        "op" : "none",
                        "ns" : "",
                        "query" : {

                        },
                        "numYields" : 0,
                        "locks" : {

                        },
                        "waitingForLock" : false,
                        "lockStats" : {

                        }
                },
                {
                        "desc" : "conn2607",
                        "threadId" : "140323415066368",
                        "connectionId" : 2607,
                        "client" : "172.16.36.87:37804",
                        "appName" : "MongoDB Shell",
                        "active" : true,
                        "opid" : 9039588,
                        "secs_running" : 0,
                        "microsecs_running" : NumberLong(63),
                        "op" : "command",
                        "ns" : "admin.",
                        "query" : {
                                "currentOp" : 1
                      },
                        "numYields" : 0,
                        "locks" : {

                        },
                        "waitingForLock" : false,
                        "lockStats" : {

                        }
                }
        ],
        "ok" : 1
    },
  ...
}
```


NOTE

- **client**: IP address of the client that sends the request
- **opid**: unique operation ID
- **secs_running**: elapsed time for execution, in seconds. If the returned value of this field is too large, check whether the request is reasonable.
- **microsecs_running**: elapsed time for execution, in seconds. If the returned value of this field is too large, check whether the request is reasonable.
- **op**: operation type. The value can be **query**, **insert**, **update**, **delete**, or **command**.
- **ns**: target collection
- For details, see the **db.currentOp()** command in **official document**.

3. Based on the command output, check whether there are requests that take a long time to process.

If the CPU usage is low while services are being processed but then becomes high during just certain operations, analyze the requests that take a long time to execute.

If an abnormal query is found, find the **opid** corresponding to the operation and run **db.killOp(*opid*)** to kill it.

Analyzing Slow Queries

Slow query profiling is enabled for DDS by default. The system automatically records any queries whose execution takes longer than 500 ms to the **system.profile** collection in the corresponding database. You can:

1. Connect to an instance using Mongo Shell.

   To access an instance from the Internet

   For details, see

   - **Connecting to a Cluster Instance over a Public Network**
   - **Connecting to a Replica Set Instance over a Public Network**
   - **Connecting to a Single Node Instance over a Public Network**

   To access an instance that is not publicly accessible

   For details, see

   - **Connecting to a Cluster Instance over a Private Network**
   - **Connecting to a Replica Set Instance over a Private Network**
   - **Connecting to a Single Node Instance over a Private Network**
2. Select a specific database (using the **test** database as an example):

   use test
3. Check whether slow SQL queries have been collected in **system.profile**.

   show collections;

   - If the command output includes **system.profile**, slow SQL queries have been generated. Go to the next step.

     ```
     mongos> show collections
     system.profile
     test
     ```
   - If the command output does not contain **system.profile**, no slow SQL queries have been generated, and slow query analysis is not required.

     ```
     mongos> show collections
     test
     ```
4. Check the slow query logs in the database.

   db.system.profile.find().pretty()
5. Analyze slow query logs to find the cause of the high CPU usage.

   The following is an example of a slow query log. The log shows a request that scanned the entire table, including 1,561,632 documents and without using a search index.

   ```
   {
   	"op" : "query",
   	"ns" : "taiyiDatabase.taiyiTables$10002e",
   	"query" : {
   		"find" : "taiyiTables",
   		"filter" : {
   			"filed19" : NumberLong("852605039766")
   		},
   ```



```
        "shardVersion" : [
                Timestamp(1, 1048673),
                ObjectId("5da43185267ad9c374a72fd5")
        ],
        "chunkId" : "10002e"
},
"keysExamined" : 0,
"docsExamined" : 1561632,
"cursorExhausted" : true,
"numYield" : 12335,
"locks" : {
        "Global" : {
                "acquireCount" : {
                        "r" : NumberLong(24672)
                }
        },
        "Database" : {
                "acquireCount" : {
                        "r" : NumberLong(12336)
                }
        },
        "Collection" : {
                "acquireCount" : {
                        "r" : NumberLong(12336)
                }
        }
},
"nreturned" : 0,
"responseLength" : 157,
"protocol" : "op_command",
"millis" : 44480,
"planSummary" : "COLLSCAN",
"execStats" : {
      "stage" : "SHARDING_FILTER",
        "nReturned" : 0,
        "executionTimeMillisEstimate" : 43701,
        "works" : 1561634,
        "advanced" : 0,
        "needTime" : 1561633,
        "needYield" : 0,
        "saveState" : 12335,
        "restoreState" : 12335,
        "isEOF" : 1,
        "invalidates" : 0,
        "chunkSkips" : 0,
        "inputStage" : {
                "stage" : "COLLSCAN",
                "filter" : {
                        "filed19" : {
                                "$eq" : NumberLong("852605039766")
                        }
                },
                "nReturned" : 0,
                "executionTimeMillisEstimate" : 43590,
                "works" : 1561634,
                "advanced" : 0,
                "needTime" : 1561633,
                "needYield" : 0,
                "saveState" : 12335,
                "restoreState" : 12335,
                "isEOF" : 1,
                "invalidates" : 0,
                "direction" : "forward",
                "docsExamined" : 1561632
        }
},
"ts" : ISODate("2019-10-14T10:49:52.780Z"),
"client" : "172.16.36.87",
"appName" : "MongoDB Shell",
```



```
        "allUsers" : [
                {
                        "user" : "__system",
                        "db" : "local"
                }
        ],
        "user" : "__system@local"
}
```

The following stages can be causes for a slow query:

- **COLLSCAN** involves a full collection (full table) scan.

  When a request (such as query, update, and delete) requires a full table scan, a large amount of CPU resources are occupied. If you find **COLLSCAN** in the slow query log, a full table scan was performed and that occupy a lot of CPU resources.

  If such requests are frequent, create indexes for the fields to be queried.

- **docsExamined** involves a full collection (full table) scan.

  You can view the value of **docsExamined** to check the number of documents scanned. A larger value indicates a higher CPU usage.

- **IXSCAN** and **keysExamined** scan indexes.

  NOTE

  - An excessive number of indexes can affect the write and update performance.
  - If your application has more write operations, creating indexes may increase write latency.

  You can view the value of **keysExamined** to see how many indexes are scanned in a query. A larger value indicates a higher CPU usage.

  If an index is not properly created or there are many matching results, the CPU usage does not decrease greatly and the execution speed is slow.

  Example: For the data of a collection, the number of values of the **a** field is small (only **1** and **2**), but the **b** field has more values.

  ```
  { a: 1, b: 1 }
  { a: 1, b: 2 }
  { a: 1, b: 3 }
  ......
  { a: 1, b: 100000}
  { a: 2, b: 1 }
  { a: 2, b: 2 }
  { a: 2, b: 3 }
  ......
  { a: 1, y: 100000}
  ```

  The following shows how to implement the {a: 1, b: 2} query.

  ```
  db.createIndex({a: 1}): The query is not effective because the a field has too many same values.
  db.createIndex({a: 1, b: 1}): The query is not effective because the a field has too many same values.
  db.createIndex({b: 1}): The query is effective because the b field has a few same values.
  db.createIndex({b: 1, a: 1}): The query is not effective because the a field has a few same values.
  ```

  For the differences between {a: 1} and {b: 1, a: 1}, see the **official documents**.

- **SORT** and **hasSortStage** may involve sorting a large amount of data.

If the value of the **hasSortStage** parameter in the **system.profile** collection is **true**, the query request involves sorting. If the sorting cannot be implemented through indexes, the query results are sorted, and sorting is a CPU intensive operation. In this scenario, you need to create indexes for fields that are frequently sorted.

If the **system.profile** collection contains **SORT**, you can use indexing to improve sorting speed.

Other operations, such as index creation and aggregation (combinations of traversal, query, update, and sorting), also apply to the above mentioned scenarios because they are also CPU intensive operations. For more information about profiling, see **official documents**.

Analysis Capability

After the analysis and optimization of the requests that are being executed and slow requests, all requests use proper indexes, and the CPU usage becomes stable. If the CPU usage remains high after the analysis and troubleshooting, the current instance may have reached the performance bottleneck and cannot meet service requirements. In this case, you can perform the following operations to solve the problem:

1. View monitoring information to analyze instance resource usage. For details, see **Viewing Monitoring Metrics**.
2. Change the DDS instance class or add shard nodes.

4.5 High Storage Usage

Scenarios

If the storage usage of a DDS instance is too high or the storage is fully used, the instance becomes unavailable.

This section describes how to analyze and fix high storage usage.

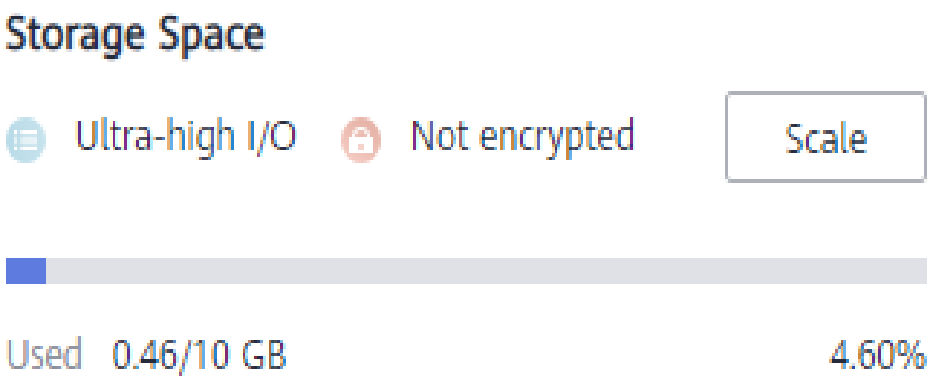
Checking the Storage Usage

DDS provides the following two methods to check the storage usage of an instance:

1. Check the storage usage on the DDS console.

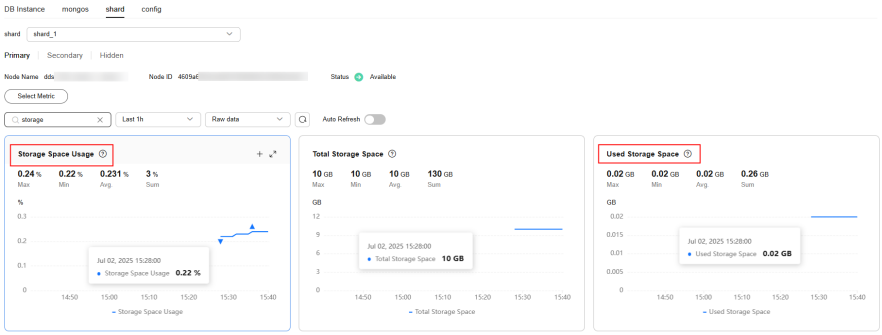
   You can log in to the DDS console and click the instance. On the **Basic Information** page, you can view the storage space of the instance in the **Storage Space** area.

Figure 4-3 Checking the storage usage



2. View the monitoring metrics (storage usage and used storage).
   To view monitoring metrics, see **Viewing Monitoring Metrics**.

Figure 4-4 Checking the storage usage



Solution

1. For cluster instances, data may be unevenly distributed because the database collection is not properly sharded. As a result, the storage usage is high.
   Shard the database collection properly.
2. As service data increases, the original database storage is insufficient. You can expand the storage space to fix this problem.
   - To scale up storage for cluster instances, see **Scaling Up a Cluster Instance**.
   - To scale up storage for replica set instances, see **Scaling Up a Replica Set Instance**.
   - To scale up storage for single node instances, see **Scaling Up a Single Node Instance**.

   If the storage space has reached the upper limit of your instance class, change the instance class first.
   - To change the cluster instance class, see **Changing a Cluster Instance Class**.
   - To change the replica set instance class, see **Changing a Replica Set Instance Class**.

  - To change the single node instance class, see **Changing a Single Node Instance Class**.

3. If a large number of expired files occupy the storage space, delete the expired files in time. For example, if the entire database is no longer used, run **dropDatabase** to delete it.
4. The background data processing mechanism is faulty.

   Operations such as write, update, and delete (including index insert and delete) are actually converted to write operations in the background. When data of an instance in use is deleted, the disk space is not reclaimed. Such unreclaimed disk space is called disk fragments. When new data is inserted, these fragments are reused without applying for new disk space. Different underlying storage engines (RocksDB and WiredTiger) vary according to specific scenarios.

   After deleting data, RocksDB directly converts the **delete** operation to append write. After a certain amount of redundant data is accumulated, the background compact thread is automatically triggered to merge and aggregate data of multiple versions to release redundant disk space. You are advised to wait for the system to automatically reclaim the disk space. If the storage usage is high and close to the **read-only** threshold, submit a service ticket.

   After deleting data, WiredTiger merges and aggregates data of multiple versions, causing disk space fragments. However, WiredTiger does not return the disk space to the operating system. WiredTiger marks the disk space for subsequent writes of the current collection, the reserved disk space is preferentially used for subsequent writes of the collection. To release the disk space, run the **compact** command. (Note: This command blocks normal services and is disabled by default.)

4.6 High Memory Usage

Scenarios

If the memory usage of a DDS instance reaches 90% and the swap space usage exceeds 5%, the system responds slowly and even out of memory (OOM) may occur.

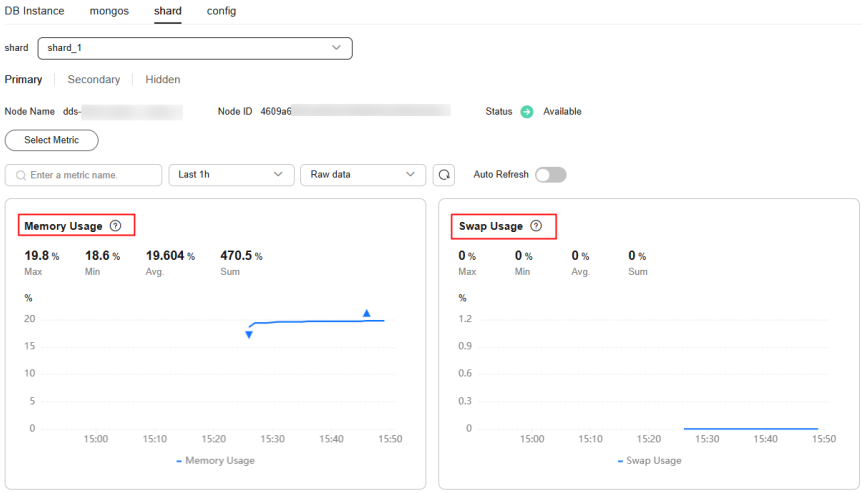
This section describes how to fix high memory usage of DB instances.

Viewing the Memory Usage

You can view the monitoring metrics (memory usage and swap usage) to learn the memory usage of instances.

For details, see **Viewing DDS Metrics**.

Figure 4-5 Memory and swap usage



NOTE

By default, 50% memory is reserved, so if the memory usage is 50% but the instance is unloaded, this is normal and you can ignore it.

Solution

1. Control the number of concurrent connections. When connecting to databases, calculate the number of clients and the size of the connection pool configured for each client. The total number of connections cannot exceed 80% of the maximum number of connections supported by the current instance. If there are too many connections, the memory and multi-thread context overhead increases, affecting the delay in request processing.
2. Configure a connection pool. The maximum number of connections in a connection pool is 200.
3. Reduce the memory overhead of a single request. For example, create indexes to reduce collection scanning and memory sorting.
4. If the number of connections remains unchanged but the memory usage keeps increasing, upgrade the memory configuration to prevent system performance deterioration caused by memory overflow and large-scale cache clearing.
   - To change cluster instance memory, see **Changing a Cluster Instance Class**.
   - To change replica set instance memory, see **Changing a Replica Set Instance Class**.
   - To change single node instance memory, see **Changing a Single Node Instance Class**.

4.7 Load Imbalance of Cluster Instances

Scenarios

It is common that load is imbalanced between shard nodes in a cluster instance. If the shard key is incorrectly selected, no chunk is preset, and the load balancing speed between shard nodes is lower than the data insertion speed, load imbalance may occur.

This section describes how to fix load imbalance.

Fault Locating

Step 1 **Connect to a database from the client.**

Step 2 Run the following command to check the shard information:

sh.status()

```
mongos> sh.status()
 \--- Sharding Status ---
  sharding version: {
        "_id" : 1,
        "minCompatibleVersion" : 5,
        "currentVersion" : 6,
        "clusterId" : ObjectId("60f9d67ad4876dd0fe01af84")
  }
  shards:
        {  "_id" : "shard_1",  "host" : "shard_1/172.16.51.249:8637,172.16.63.156:8637",  "state" : 1 }
        {  "_id" : "shard_2",  "host" : "shard_2/172.16.12.98:8637,172.16.53.36:8637",  "state" : 1 }
  active mongoses:
        "4.0.3" : 2
  autosplit:
        Currently enabled: yes
  balancer:
        Currently enabled:  yes
        Currently running:  yes
        Collections with active migrations:
              test.coll started at Wed Jul 28 2021 11:40:41 GMT+0000 (UTC)
        Failed balancer rounds in last 5 attempts:  0
        Migration Results for the last 24 hours:
              300 : Success
  databases:
        {  "_id" : "test",  "primary" : "shard_2",  "partitioned" : true,  "version" : {  "uuid" : UUID("d612d134-
a499-4428-ab21-b53e8f866f67"),  "lastMod" : 1 } }
              test.coll
                    shard key: { "_id" : "hashed" }
                    unique: false
                    balancing: true
                    chunks:
                          shard_1 20
                          shard_2 20
```

- **databases** lists databases for which you enable **enableSharding**.
- **test.coll** is the collection namespace. **test** indicates the name of the database where the collection is located, and **coll** indicates the name of the collection for which sharding is enabled.
- **shard key** is the shard key of the previous collection. **_id**: indicates that the shard is hashed based on **_id**. **_id: -1** indicates that the shard is sharded based on the range of **_id**.

- **chunks** indicates the distribution of shards.

Step 3 Analyze the shard information based on the query result in **Step 2**.

1. If no shard information is queried, the collections are not sharded.

   Run the following command to enable sharding:

   ```
   mongos> sh.enableSharding("<database>")
   mongos> use admin
   mongos> db.runCommand({shardcollection:"<database>.<collection>",key:{"keyname":<value> }})
   ```

2. If an improper shard key is selected, the load may be imbalanced. For example, if a large number of requests are processed on a range of shards, the load on these shards is heavier than other shards, causing load imbalance.

   You can redesign the shard key, for example, changing ranged sharding to hashed sharding.

   ```
   mongos> db.runCommand({shardcollection:"<database>.<collection>",key:{"keyname":<value> }})
   ```

   NOTE

   - If a sharding mode is determined, it cannot be changed easily. The sharding mode must be fully considered in the design phase.

3. If a large amount of data is inserted and the data volume exceeds the load capacity of a single shard, shard imbalance occurs and the storage usage of the primary shard is too high.

   You can run the following command to check the network connection of the server and check whether the amount of data transmitted by each network adapter reaches the upper limit.

   ```
   sar -n DEV 1  //1 is the interval.
   Average: IFACE  rxpck/s  txpck/s   rxkB/s   txkB/s rxcmp/s txcmp/s rxmcst/s %ifutil
   Average:    lo  1926.94  1926.94  25573.92  25573.92    0.00    0.00    0.00    0.00
   Average:  A1-0     0.00     0.00      0.00      0.00    0.00    0.00    0.00    0.00
   Average:  A1-1     0.00     0.00      0.00      0.00    0.00    0.00    0.00    0.00
   Average:  NIC0     5.17     1.48      0.44      0.92    0.00    0.00    0.00    0.00
   Average:  NIC1     0.00     0.00      0.00      0.00    0.00    0.00    0.00    0.00
   Average:  A0-0  8173.06 92420.66  97102.22 133305.09    0.00    0.00    0.00    0.00
   Average:  A0-1 11431.37  9373.06 156950.45    494.40    0.00    0.00    0.00    0.00
   Average:  B3-0     0.00     0.00      0.00      0.00    0.00    0.00    0.00    0.00
   Average:  B3-1     0.00     0.00      0.00      0.00    0.00    0.00    0.00    0.00
   ```

   NOTE

   - **rxkB/s** is the number of KBs received per second.
   - **txkB/s** is the number of KBs sent per second.

   After the check is complete, press **Ctrl+Z** to exit.

   If the network load is too high, analyze MQL statements, optimize the roadmap, reduce bandwidth consumption, and increase specifications to expand network throughput.

   - Check whether there are sharded collections that do not carry ShardKey. In this case, requests are broadcast, which increases the bandwidth consumption.
   - Control the number of concurrent threads on the client to reduce the network bandwidth traffic.
   - If the problem persists, **increase instance specifications** in a timely manner. High-specification nodes can provide higher network throughput.

----End

4.8 Slow Request Locating

Scenarios

In a given service scenario, the query performance depends on the design of the architecture, databases, collections, and indexes. A good design can improve the query performance. On the contrary, a large number of slow queries (statements that take a long time to execute) may occur, which deteriorates system performance.

This document describes the causes and solutions of slow queries.

Fault Locating

DDS allows you to **view slow query logs** on the console. You can start from the slowest operation recorded in the log and optimize the operations one by one.

- If a query takes longer than 1s, the corresponding operation may be abnormal. You need to analyze the problem based on the actual situation.
- If a query takes longer than 10s, the operation needs to be optimized.

  NOTE

  If an aggregate operation takes more than 10s, it is normal.

Analysis Method

Step 1 Connect to the database.

- To connect to a cluster instance, see **Connecting to a Cluster Instance**.
- To connect to a replica set instance, see **Connecting to a Replica Set Instance**.
- For details about how to connect to a single node instance, see **Connecting to a Single Node Instance**.

Step 2 Run the following command to check the execution plan of a slow query:

explain()

Example:

```
db.test.find({"data_id" : "ae4b5769-896f-465c-9fbd-3fd2f3357637"}).explain();
db.test.find({"data_id" : "775f57c2-b63e-45d7-b581-3822dba231b4"}).explain("executionStats");
```

A covered query does not need to read a document, but directly returns a result from an index, which is very efficient. You can use covering indexes as much as possible. If the output of explain() shows that indexOnly is true, the query is covered by an index.

Step 3 Parse the execution plan.

1. Check the execution time.

   The smaller the values of the following parameters, the better the performance: **executionStats.executionStages.executionTimeMillisEstimate**

and **executionStats.executionStages.inputStage.executionTimeMillisEstimate**

Table 4-3 Parameter description

Parameter	Description
executionStats.executionTimeMillis	Execution plan selection and execution time
executionStats.executionStages.executionTimeMillisEstimate	Execution completion time of the execution plan
executionStats.executionStages.inputStage.executionTimeMilli-sEstimate	Execution completion time of the sub-phase of the execution plan

2. Check the number of scanned records.

   If the three items in **Table 4-4** have the same value, the query performance is the best.

   Table 4-4 Parameter description

Parameter	Description
executionStats.nReturned	Number of documents matching the search criteria
executionStats .totalKeysExamined	Number of rows scanned through indexes
executionStats .totalDocsExamined	Number of scanned documents

3. Check the stage status.

   The combinations of stage statuses with better performance are as follows:

   - Fetch+IDHACK
   - Fetch+ixscan,
   - Limit+ (Fetch+ixscan)
   - PROJECTION+ixscan

   Table 4-5 Status description

Status Name	Description
COLLSCAN	Full table scan
SORT	In-memory sorting
IDHACK	_id-based query

Status Name	Description
TEXT	Full-text index
COUNTSCAN	Number of unused indexes
FETCH	Index scanning
LIMIT	Using Limit to limit the number of returned records
SUBPLA	$or query stage without using an index
PROJECTION	Number of used indexes
COUNT_SCAN	Number of used indexes

----End

Optimization Plan

- For queries without indexes, create indexes based on the search criteria.
- Hash indexes can be created for point queries.
- Create composite indexes for multi-field queries where a single field is highly repeated.
- Create an ascending or descending index for range lookups with ordered result sets.
- Compound indexes are those indexes sort query results by prefix, so the sequence of query conditions must be the same as that of index fields.
- For partitioned collections (tables) and large collections (with more than 100,000 records), do not use fuzzy query (or do not use LIKE) for tables with a large amount of data. As a result, a large number of records are scanned. You can query data based on the index field, filter out small collections, and then perform fuzzy queries.
- Do not use $not. MongoDB does not index missing data. The $not query requires that all records be scanned in a single result collection. If $not is the only query condition, a full table scan will be performed on the collection.
- If you use $and, put the conditions with the fewest matches before other conditions. If you use $or, put the conditions with the more matches first.
- Check the performance baseline of instance specifications and analyze whether the current service requirements can be met. If the performance bottleneck of the current instance is reached, upgrade the instance specifications in a timely manner.

4.9 Statement Optimization

DDS is inherently a NoSQL database with high performance and strong extensibility. Similar to relational databases, such as RDS for MySQL, RDS for SQL Server, and Oracle, DDS instance performance may also be affected by database design, statement optimization, and index creation.

The following provides suggestions for improving DDS performance in different dimensions:

Creating Databases and Collections

- Use short field names to save storage space. Different from an RDS database, each DDS document has its field names stored in the collection. Short name is recommended.
- Limit the number of documents in a collection to avoid the impact on the query performance. Archive documents periodically if necessary.
- Each document has a default **_id**. Do not change the value of this parameter.
- Capped collections have a faster insertion speed than other collections and can automatically delete old data. You can create capped collections to improve performance based on your service requirements.

Query Operations

Indexes

- Create proper number of indexes for frequently queried fields based on service requirements. Indexes occupy some storage space, and the insert and indexing operations consume resources. It is recommended that the number of indexes in each collection should not exceed 5.
- If data query is slow due to lack of indexes, create proper indexes for frequently queried fields.
- For a query that contains multiple shard keys, create a compound index that contains these keys. The order of shard keys in a compound index is important. A compound index support queries that use the leftmost prefix of the index, and the query is only relevant to the creation sequence of indexes.
- TTL indexes can be used to automatically filter out and delete expired documents. The index for creating TTL must be of type date. TTL indexes are single-field indexes.
- You can create field indexes in a collection. However, if a large number of documents in the collection do not contain key values, you are advised to create sparse indexes.
- When you create text indexes, the field is specified as **text** instead of **1** or **-1**. Each collection has only one text index, but it can index multiple fields.

Command usage

- The findOne method returns the first document that satisfies the specified query criteria from the collection according to the natural order. To return multiple documents, use this method.
- If the query does not require the return of the entire document or is only used to determine whether the key value exists, you can use **$project** to limit the returned field, reducing the network traffic and the memory usage of the client.
- In addition to prefix queries, regular expression queries take longer to execute than using selectors, and indexes are not recommended.
- Some operators that contain **$** in the query may deteriorate the system performance. The following types of operators are not recommended in services. $or, $nin, $not, $ne, and $exists.

Table 4-6 Operator description

Operator	Description
$or	The times of queries depend on the number of conditions. It is used to query all the documents that meet the query conditions in the collection. You are advised to use $in instead.
$nin	Matches most of indexes, and the full table scan is performed.
$not	The query optimizer may fail to match a specific index, and the full table scan is performed.
$ne	Selects the documents where the value of the field is not equal to the specified value. The entire document is scanned.
$exists	Matches each document that contains the field.

For more information, see **official MongoDB documents**.

Precautions

- Indexes cannot be used in operators $where and $exists.
- If the query results need to be sorted, control the number of result sets.
- If multiple field indexes are involved, place the field used for exact match before the index.
- If the key value sequence in the search criteria is different from that in the compound index, DDS automatically changes the query sequence to the same as index sequence.
  - Modification operation

    Modify a document by using operators can improve performance. This method does not need to obtain and modify document data back and forth on the server, and takes less time to serialize and transfer data.
  - Batch insert

    Batch insert can reduce the number of times data is submitted to the server and improve the performance. The BSON size of the data submitted in batches cannot exceed 48 MB.
  - Aggregate operation

    During aggregation, $match must be placed before $group to reduce the number of documents to be processed by the $group operator.

CAUTION

Improper optimization of slow queries may cause service exceptions.

5 Instance Lifecycle Management

5.1 Instance Statuses

Scenarios

The status of a DB instance indicates the health of the DB instance. You can check the status of a DB instance either on the management console or by calling APIs.

DB Instance Status

Table 5-1 Status and description

Status	Description
Available	A DB instance is running properly.
Abnormal	A DB instance is faulty.
Creating	A DB instance is being created.
Creation failed	A DB instance fails to be created.
Backing up	An instance backup is being created.
Restarting	A DB instance is being restarted because of a modification that requires restarting it for the modification to take effect.
Switchover in progress	The primary and secondary nodes of the replica set instance or the primary and secondary shard or config nodes of a cluster instance are being switched over.
Adding node	shard or dds mongos nodes are being added to a DDS cluster instance.
Deleting node	The node that failed to be added is being deleted.
Scaling up	The storage space of instance nodes is being expanded.

Status	Description
Changing instance class	The CPU or memory of a DB instance is being changed.
Changing to yearly/monthly	The billing mode is being changed from pay-per-use to yearly/monthly.
Checking restoration	The backup of the current DB instance is being restored to a new DB instance.
Restoring	The backup is being restored to the existing DB instance.
Restore failed	Restoring to the existing DB instance failed.
Switching SSL	The SSL channel is being enabled or disabled.
Querying original slow query logs	Show Original Log is being enabled or disabled.
Changing private IP address	The private IP address of a node is being changed.
Changing port	The DB instance port is being changed.
Changing a security group	The security group is being changed.
Frozen	DB instances are frozen when there is no balance in the account.
Minor version upgrade	The minor version upgrade is in progress.
Checking changes	Status of a yearly/monthly instance when the billing mode is being changed.

Parameter Template Status

Table 5-2 Status and description

Status	Description
In-Sync	A database parameter change has taken effect.
Available	Parameters change. Pending restart

5.2 Exporting Instance Information

Scenarios

On the DDS console, you can export information about all DDS instances or information about a specified instance.

Exporting Information of All Instances

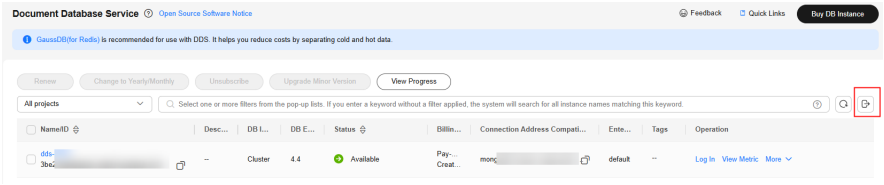
Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

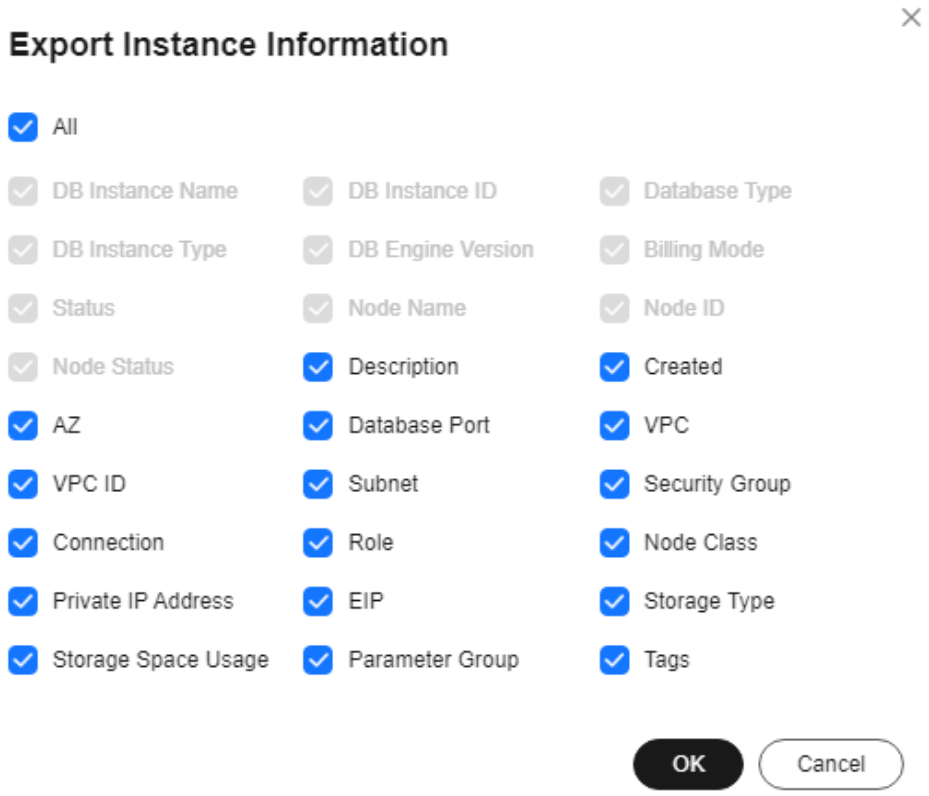
Step 4 On the **Instances** page, click  in the upper right corner of the instance list.

Figure 5-1 Exporting the instance information



Step 5 In the pop-up box, select the desired items and click **OK**.

Figure 5-2 Export Instance Information



Step 6 View the .xls file exported to your local PC.

----End

Exporting Information of a Specified Instance

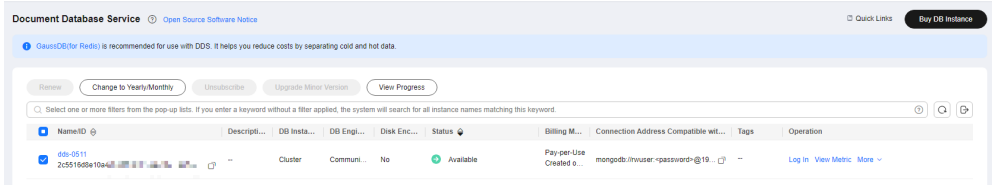
Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

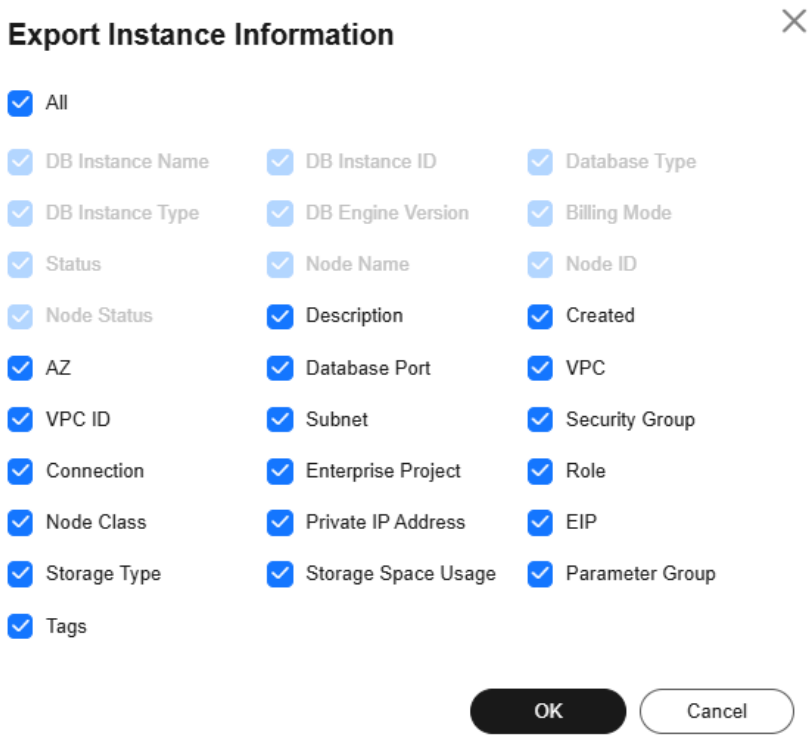
Step 4 On the **Instances** page, select the instance and click in the upper right corner of the instance list.

Figure 5-3 Exporting required instance information



Step 5 In the pop-up box, select the desired items and click **OK**.

Figure 5-4 Export Instance Information



Step 6 View the .xls file exported to your local PC.

----End

5.3 Restarting an Instance or a Node

Scenarios

You may need to occasionally restart an instance to perform routine maintenance. For example, when the number of connections reaches the upper limit, the instance performance is poor, or after modifying certain parameters, you may need to restart your instance to apply the modifications.

Precautions

- You can restart an instance only when its status is **Available**.
- Restarting an instance will interrupt services. Exercise caution when performing this operation.
- This instance is not available when it is being restarted. Restarting an instance will clear the cached memory in it. You are advised to restart it during off-peak hours.
- If you restart a cluster or replica set instance, all nodes in the instance are also restarted.

- You can restart a cluster instance or any dds mongos, shard, config node, or read replica in the cluster instance. During the restart, the node cannot be accessed.
- You can restart a replica set instance. During the restart, the instance cannot be accessed.
- You can restart any read replica in a replica set instance. During the restart, the node cannot be accessed.
- You can forcibly restart an abnormal node in a DB instance. The node cannot be accessed during the restart.
- After a replica set instance is restarted, the node roles may change.
- It takes less than 30 seconds to start a mongod or dds mongos process. If there are a large number of collections (more than 10,000), it may take several minutes to start the Mongod process. Before the startup is complete, the corresponding node cannot be connected. You are advised to limit the number of collections to less than 10,000 to avoid excessive service loss due to long-time startup.

Restarting an Instance

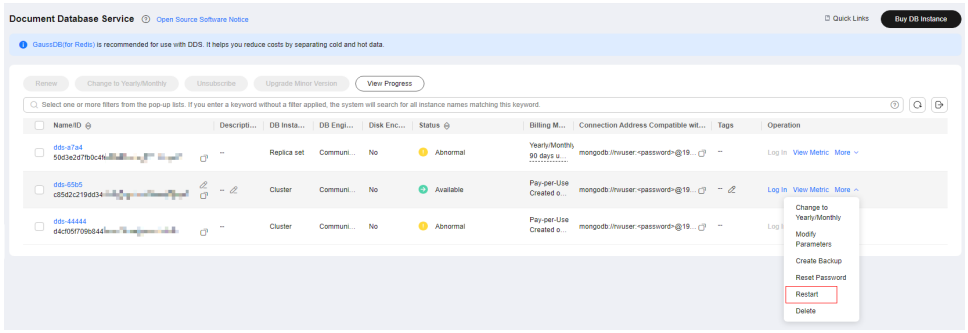
Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

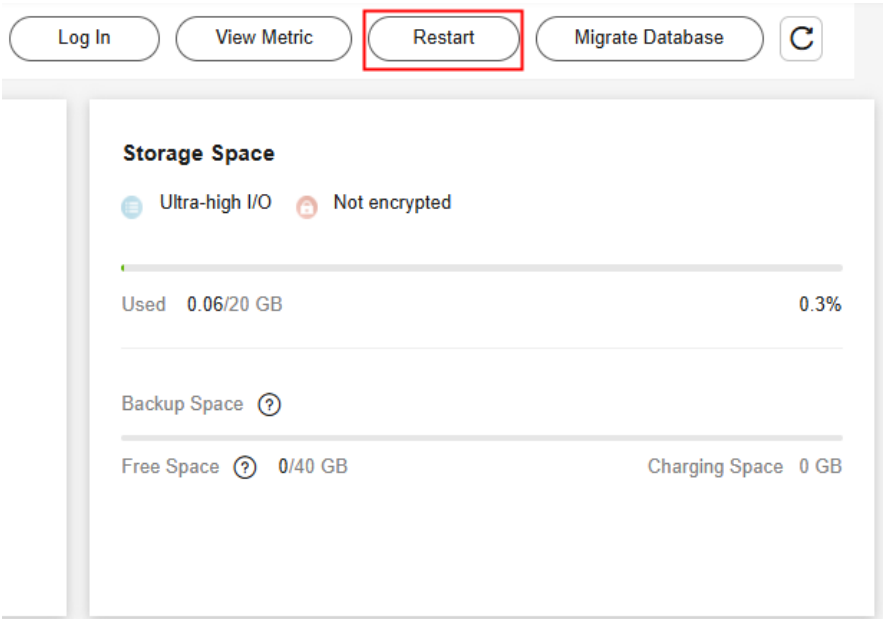
Step 4 On the **Instances** page, locate the instance and in the **Operation** column, choose **More** > **Restart**.

Figure 5-5 Restarting an instance



Alternatively, click the instance name and on the displayed **Basic Information** page, click **Restart** in the upper right corner of the page.

Figure 5-6 Restarting an instance



Step 5 If you have enabled operation protection, click **Start Verification** in the **Restart DB Instance** dialog box. On the displayed page, click **Send Code**, enter the verification code, and click **Verify**. The page is closed automatically.

Step 6 In the displayed dialog box, click **Yes**.

Step 7 View the instance status.

On the **Instances** page, the instance status is **Restarting**.

----End

Restarting a Cluster Node

Step 1 Log in to the management console.

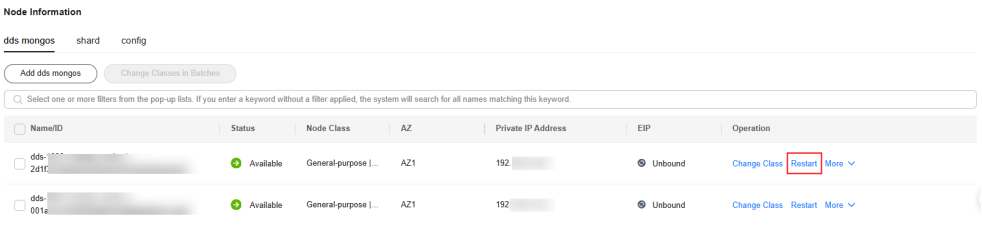
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the cluster instance name.

Step 5 In the **Node Information** area on the **Basic Information** page, click the **dds mongos**, **shard**, or **config** tab, locate a node, and in the **Operation** column, click **Restart**.

Figure 5-7 Restarting a dds mongos node



Step 6 In the displayed dialog box, click **Yes**.

Step 7 View the node status.

When one node status is **Restarting**, other nodes of the instance cannot be restarted.

----End

Restarting a Read Replica of a Replica Set Instance

Step 1 Log in to the management console.

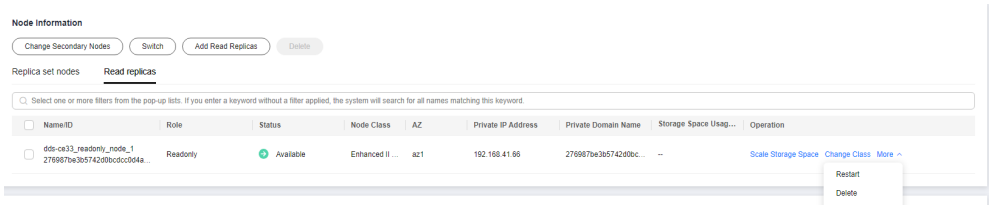
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the replica set instance.

Step 5 In the **Node Information** area on the **Basic Information** page, click the **Read replicas** tab, locate the read replica to be restarted, and click **More** in the **Operation** column.

Figure 5-8 Read replicas



Step 6 Select **Restart**.

Step 7 In the displayed dialog box, click **Yes** to restart the read replica.

Step 8 View the status of the read replica.

When one node status is **Restarting**, other nodes of the instance cannot be restarted.

----End

Forcibly Restarting an Abnormal Node

Step 1 Log in to the management console.

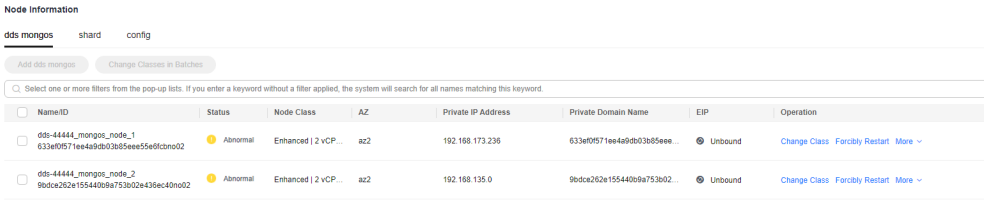
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the target DB instance and click its name.

Step 5 In the **Node Information** area on the **Basic Information** page, click **Forcibly Restart** in the **Operation** column of the target abnormal node.

Figure 5-9 Selecting an abnormal node



Step 6 In the displayed dialog box, click **Yes** to restart the abnormal node.

Step 7 View the status of the node.

When one node status is **Restarting**, other nodes of the instance cannot be restarted.

----End

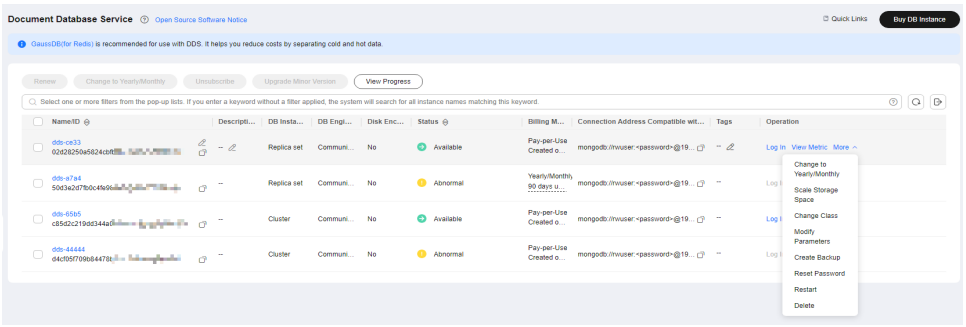
Restarting Nodes in a Replica Set Instance One by One

Step 1 Click  in the upper left corner and select a region and a project.

Step 2 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

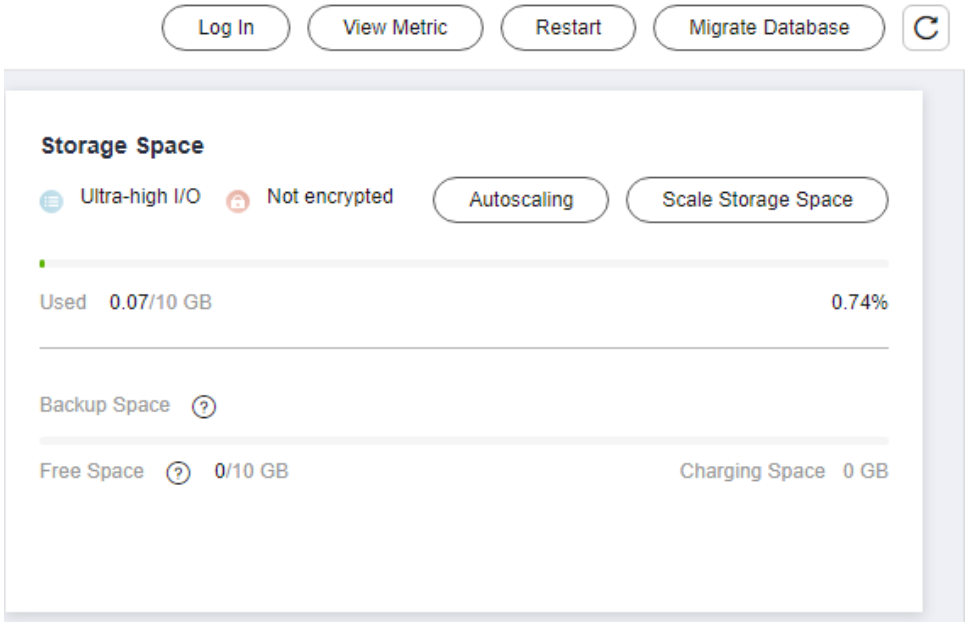
Step 3 On the **Instances** page, locate the replica set instance and in the **Operation** column, choose **More** > **Restart**.

Figure 5-10 Restarting a replica set instance



Alternatively, click the replica set instance name and on the displayed **Basic Information** page, click **Restart** in the upper right corner of the page.

Figure 5-11 Restarting a replica set instance



Step 4 In the displayed dialog box, select **Restart nodes one by one**.

Step 5 Click **Yes** to restart the replica set instance nodes one by one.

Step 6 Check the DB instance status.

On the **Instances** page, the instance status is **Restarting**. If nodes in a replica set instance are restarted one by one, a primary/secondary switchover is triggered.

----End

5.4 Deleting Pay-per-Use Instances

Scenarios

You can delete a DB instance billed on a pay-per-use basis on the **Instances** page to release resources. After you delete an instance, all of the nodes for that instance are deleted along with it.

Precautions

- To delete an instance billed on a yearly/monthly basis, you need to unsubscribe from the order. For details, see .
- After you delete the instance, all its data and all automated backups are automatically deleted as well and cannot be restored. Exercise caution when performing this operation.
- By default, all manual backups are retained in DDS. You can use a backup to restore a deleted instance.
- If you enable operation protection to improve the security of your account and cloud products, two-factor authentication is required for sensitive operations. For details about how to enable operation protection, see Operation Protection in *Identity and Access Management User Guide*.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the instance and choose **More** > **Delete** in the **Operation** column.

Step 5 If you have enabled operation protection, click **Start Verification** in the **Delete DB Instance** dialog box. On the displayed page, click **Send Code**, enter the verification code, and click **Verify**. The page is closed automatically.

Step 6 In the displayed dialog box, click **Yes**.

----End

5.5 Recycling an Instance

5.5.1 Modifying the Recycling Policy

Scenarios

DDS allows you to move unsubscribed yearly/monthly DB instances and deleted pay-per-use DB instances to the recycle bin.

Precautions

- The recycling policy is enabled by default and cannot be disabled. Instances in the recycle bin are retained for 7 days by default, and this will not incur any charges.
- Up to 100 instances can be moved to the recycle bin. Once the recycle bin is full, you can still delete instances, but they cannot be placed in the recycle bin, so the deletions will be permanent.
- You can modify the retention period, and the changes only apply to the instances deleted after the changes, so exercise caution when performing this operation.
- Recycling and backup cannot be performed when a node is in the **UNKNOWN** state.

Procedure

Step 1 Log in to the management console.

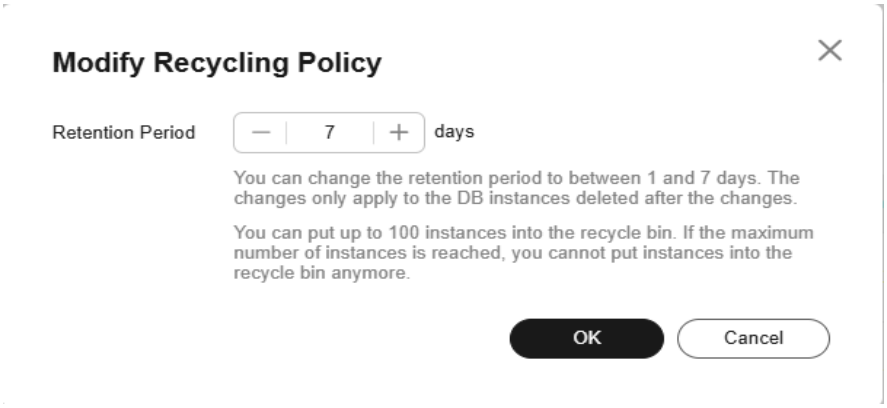
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Recycle Bin**.

Step 5 On the **Recycle Bin** page, click **Modify Recycling Policy**. In the displayed dialog box, set the retention period for the deleted DB instances (range: 1 to 7 days). Then, click **OK**.

Figure 5-12 Modify Recycling Policy



----End

5.5.2 Rebuilding an Instance

Scenarios

You can rebuild an instance from the recycle bin to restore data.

Precautions

- You can rebuild DB instances within the retention period from the recycle bin.
- After an instance is deleted, the latest automated full backup of the previous day is retained. If no automated full backup of the previous day exists, the latest automated full backup is retained. Additionally, a full backup is performed. You can select either of the backups to rebuild the instance.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Recycle Bin**.

Step 5 On the **Recycle Bin** page, locate the instance to be rebuilt and in the **Operation** column, click **Rebuild**.

Figure 5-13 Rebuilding a DB Instance

Step 6 On the displayed page, set required parameters and submit the rebuilding task. For details, see **Restoring Data to a New Instance**.

----End

6 Version Upgrade

6.1 Upgrading a Minor Engine Version

Scenarios

DDS supports minor version upgrade to improve performance, add new functions, and fix bugs. For details, see .

If the database version is a risky version, the system prompts you to upgrade a minor version.

If a new patch is released, you can click **Upgrade Minor Version** on the **Instances** page to upgrade the minor engine version. For details, see **Figure 6-1**.

If the kernel version of your instance has potential risks or major defects, has expired, or has been brought offline, the system will notify you by SMS message or email and deliver an upgrade task during the maintenance window.

Figure 6-1 Minor version upgrade

Precautions

- A DDS version cannot be downgraded, for example, from 4.0 to 3.4.
- Pay attention to patches that address issues and vulnerabilities from the open source community. When a new patch is released, install the patch in a timely manner.
- During the upgrade, your services may be intermittently interrupted once for up to 30s for each node. Ensure that your instance can be reconnected automatically or perform this operation during off-peak hours.
- DDL operations, such as **create event**, **drop event**, and **alter event**, are not allowed during the upgrade.

- Before the upgrade, ensure that no time-consuming operations, such as index creation and full table scans, are in progress on the service side. Otherwise, the upgrade may fail or services could be affected. You can run the **db.currentOp()** command or use real-time session query of DAS to check if any time-consuming requests are currently being executed.

Constraints

- Only cluster and replica set instances support minor engine version upgrade.
- This function is available for DB instances of version 3.4 or later.
- If the instance status is abnormal or the instance is being operated, the upgrade cannot be performed.
- The upgrade cannot be performed if the instance nodes are abnormal.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

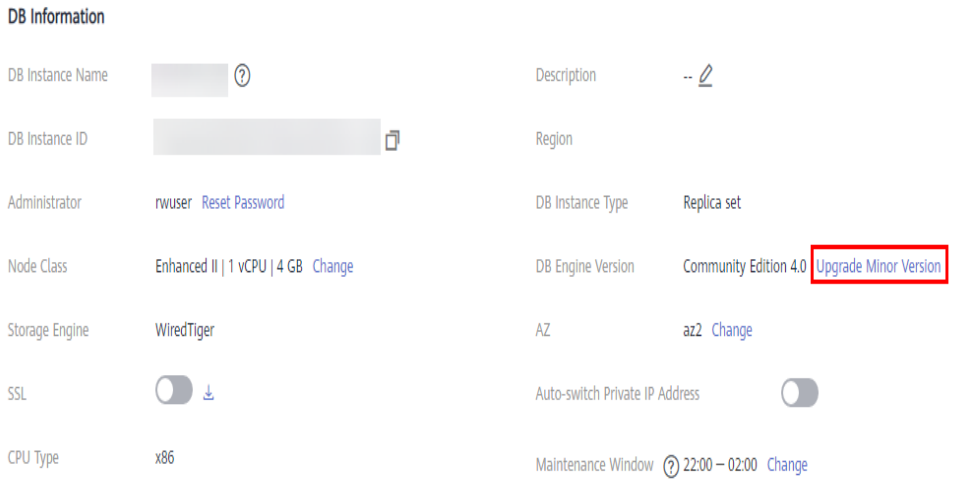
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the instance you want to upgrade and click **Upgrade Minor Version** in the **DB Engine Version** column.

Figure 6-2 Minor version upgrade

Alternatively, click the instance. In the **DB Information** area on the **Basic Information** page, click **Upgrade Minor Version** in the **DB Engine Version** field.

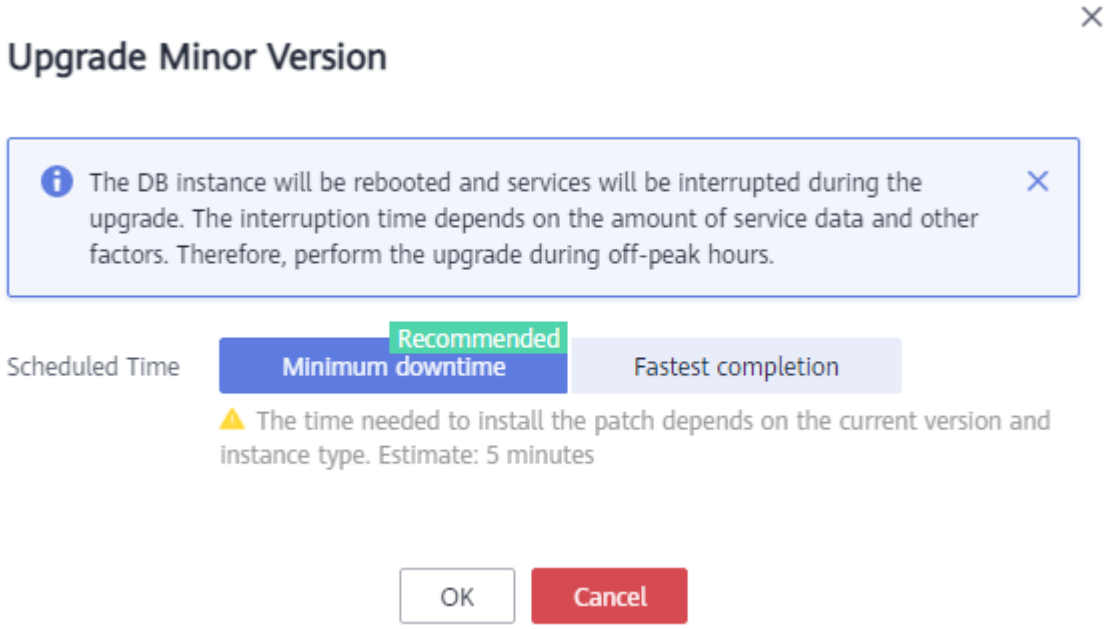
Figure 6-3 Minor version upgrade



Step 5 In the displayed dialog box, specify **Scheduled Time** based on service requirements and click **OK**. You can view the upgrade progress on the **Task Center** page.

- **Minimum downtime**: The upgrade has little impact on services.
- **Fastest completion**: The upgrade takes a relatively short time.

Figure 6-4 Selecting a scheduled time



NOTE

The time required for the upgrade varies according to the site requirement.

----End

6.2 Upgrading a Major Engine Version

Precautions

DDS does not support major engine version upgrade on the console. You can use DRS to migrate data as required.

For example, you can use DRS to migrate data from DDS 3.4 to DDS 4.0 without interrupting services.

Constraints

Before migrating data using DRS, you need to create the destination DB instance in advance.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click an instance you want to migrate. On the displayed **Basic Information** page, click **Migrate Database** in the upper right corner of the page.

Table 6-1 Database versions

Source DB Version	Destination Database Version	Migration Type
Self-built MongoDB/ Other cloud MongoDB/DDS • 3.4 • 4.0 • 4.2 • 4.4 • 5.0	DDS • 3.4 • 4.0 • 4.2 • 4.4 • 5.0	Version upgrade

NOTE

- DDS 5.0 supports replica sets only.
- Data cannot be migrated from a newer version database to an older version database.
- During the migration, if the specifications are changed or the instance is restarted, a primary/secondary switchover may occur and the migration task may be intermittently interrupted. In this case, you need to check the DRS task status in a timely manner.
- After a major version upgrade, you can change the IP address of the newer version database to the IP address of the older version database. To perform this operation, release the IP address of the older version database first. For details, see **Changing a Private IP Address**.

----End

7 Instance Modifications

7.1 Changing an Instance Name

Scenarios

DDS allows you to change the name of a DB instance for easy identification.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click next to the instance name you wish to change, enter a new name and click **OK** to apply the changes.

Alternatively, in the **DB Information** area on the **Basic Information** page, click in the **DB Instance Name** field, enter a new name and click to apply the changes.

NOTE

- The instance name can be the same as an existing instance name.
- The instance name must contain 4 to 64 characters and must start with a letter. It is case sensitive and can contain letters, digits, hyphens (-), and underscores (_). It cannot contain other special characters.

Step 5 View the results on the **Instances** page.

----End

7.2 Changing an Instance Description

Scenarios

After a DDS instance is created, you can add a description to it.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the instance you wish to edit the description for and click in the **Description** column to edit the instance description. Then, click **OK**.

Alternatively, click the target instance to go to the **Basic Information** page. In the **DB Information** area, click in the **Description** field to edit the instance description. To submit the change, click .

NOTE

The instance description can contain up to 64 characters, excluding carriage return characters and special characters >!<"&'=

Step 5 View the results on the **Instances** page.

----End

7.3 Modifying an Instance Tag

Scenarios

DDS allows you to modify tags of DB instances so that you can filter DB instances by tag.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the instance you wish to edit the tag for and click in the **Tags** column to edit the instance tag. Then, click **OK**.

Step 5 View the results on the **Instances** page.

----End

7.4 Changing the Name of the Replica Set in the Connection Address

Scenarios

You can change the name of the replica set in the connection address for a DDS DB instance to better meet your service requirements.

Precautions

- This function is available only for replica set instances.
- When you change the replica set name in the connection address of a DDS DB instance, the instance will be unavailable. Exercise caution when performing this operation.
- This operation is not allowed if the DB instance is in any of the following statuses: creating, changing instance class, changing port, restarting, or abnormal.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the target replica set instance and click its name.

Step 5 In the **Network Information** area, click next to **HA Connection Address**.

Step 6 Enter a new name and click to save the change.

NOTE

The name of the replica set in the connection address must be 3 to 128 characters long and start with a letter. It is case-sensitive and can contain only letters, digits, and underscores (_).

----End

7.5 Scaling Up Storage Space

7.5.1 Scaling Up a Cluster Instance

Scenarios

If there is not enough storage for your workloads, you can scale up the storage of your DB instance. If you scale up the storage of a DB instance, the backup space increases accordingly.

- If an instance is created before September 2023, about 5% of the disk space is reserved for the system **root** user.
- If the storage usage reaches 97% of the purchased storage space, the database is set to the read-only state. After the storage usage drops below 85%, the instance becomes readable and writable.

In addition, you can set alarm rules for the storage usage. For details, see **Configuring Alarm Rules**.

For details about the causes and solutions of insufficient storage space, see **High Storage Usage**.

Precautions

- Scaling is available when your account balance is sufficient.
- For cluster instances, only shard nodes can be scaled up. dds mongos nodes, config nodes, and read replicas cannot be scaled up.
- If you scale up a DB instance with disks encrypted, the expanded storage space will be encrypted using the original encryption key.
- An instance cannot be scaled up if it is in any of the following statuses:
  - Creating
  - Changing instance class
  - Adding node
  - Deleting node
  - Upgrading minor version
- Services are not interrupted during scaling. The storage type cannot be changed.

Billing

- A pay-per-use instance is still billed on an hourly basis after the instance is scaled up.
- If you scale up a yearly/monthly instance, you will pay price difference or get a refund.

Procedure

Step 1 Log in to the management console.

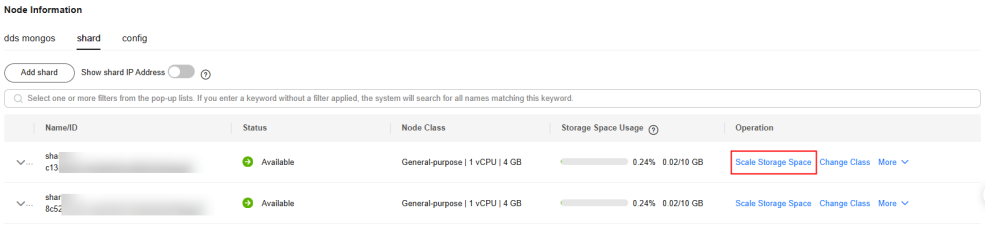
Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the cluster instance name.

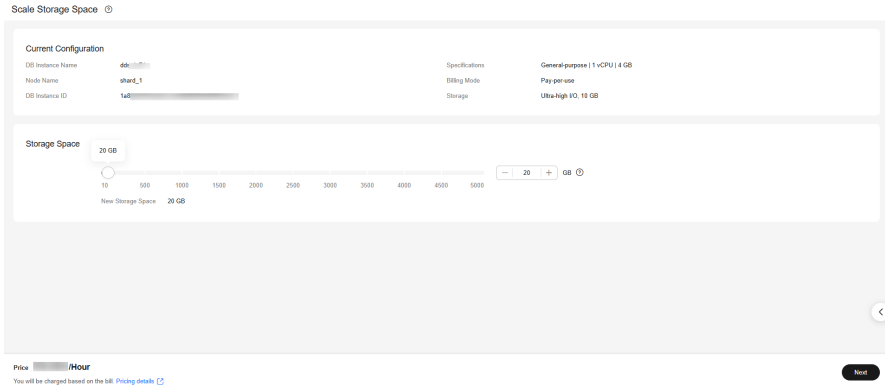
Step 5 In the **Node Information** area on the **Basic Information** page, click the **shard** tab, locate the shard node you want to scale, and click **Scale Storage Space** in the **Operation** column.

Figure 7-1 Scaling up storage space



Step 6 On the displayed page, specify the desired amount of space and click **Next**.

Figure 7-2 Scale Storage Space



Select at least 10 GB each time you scale up the storage, and the storage size must be multiples of 10 GB. If a shard node has fewer than 8 vCPUs, the maximum storage that can be scaled up to is 5000 GB. If a shard node has 8 or more vCPUs, the maximum storage that can be scaled up to is 10,000 GB.

Step 7 On the displayed page, confirm the storage space.

- For yearly/monthly DB instances
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.
  - If you do not need to modify your settings, click **Submit** to go to the payment page and complete the payment.
- For pay-per-use DB instances
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.
  - If you do not need to modify the specifications, click **Submit** to scale up the storage space.

Step 8 Check the results.

- This process takes about 3 to 5 minutes. The status of the DB instance in the instance list is **Scaling up**.

- In the upper right corner of the instance list, click to refresh the list. The instance status changes to **Available**.
- In the **Node Information** area on the **Basic Information** page, click the **shard** tab and check whether the scale up was successful.

CAUTION

If you need to scale up the storage space to more than 10 TB, submit a service ticket.

If the storage space is scaled up to more than 10 TB, the following risks may occur:

- If the purchased storage space exceeds 16,000 GB and the available storage space is 480 GB, the instance becomes read-only. After the available storage space is greater than 2,400 GB, the instance becomes readable and writable.
- If there is a large amount of data, the backup task may take a long time or even fail. In this case, the service SLA may be affected. You need to enable snapshot backup to ensure that the backup task can be executed properly. For details about how to enable snapshot backup, see **Enabling or Modifying an Automated Backup Policy**.
- If data is deleted by mistake, it takes a long time to restore a table to a specified point in time or restore a backup to a new instance, affecting the restoration efficiency.
- If the primary/secondary or read-only replication is delayed, it takes a long time to reconnect. As a result, the instance may be disconnected or fail to be reconnected.

----End

Reference

What Should I Do If Storage Usage Is Unusually High?

7.5.2 Scaling Up a Replica Set Instance

Scenarios

If there is not enough storage for your workloads, you can scale up the storage of your DB instance. If you scale up the storage of a DB instance, the backup space increases accordingly.

- If an instance is created before September 2023, about 5% of the disk space is reserved for the system **root** user.
- If the storage usage reaches 97% of the purchased storage space, the database is set to the read-only state. After the storage usage drops below 85%, the instance becomes readable and writable.

In addition, you can set alarm rules for the storage usage. For details, see **Configuring Alarm Rules**. For details about the causes and solutions of insufficient storage space, see **High Storage Usage**.

Precautions

- Scaling is available when your account balance is sufficient.
- If you scale up a DB instance with disks encrypted, the expanded storage space will be encrypted using the original encryption key.
- An instance cannot be scaled up if it is in any of the following statuses:
  - Creating
  - Changing instance class
  - Adding node
  - Deleting node
  - Upgrading minor version
  - Switchover in progress
- During scaling, services will not be interrupted, and the storage type cannot be changed.

Billing

- A pay-per-use instance is still billed on an hourly basis after the instance is scaled up.
- If you scale up a yearly/monthly instance, you will pay price difference or get a refund.

Procedure

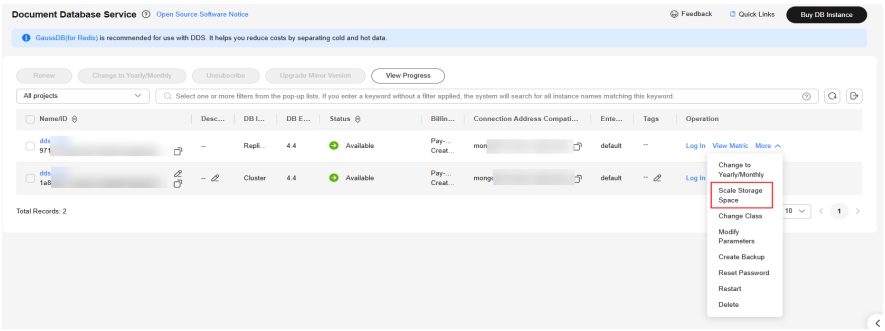
Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

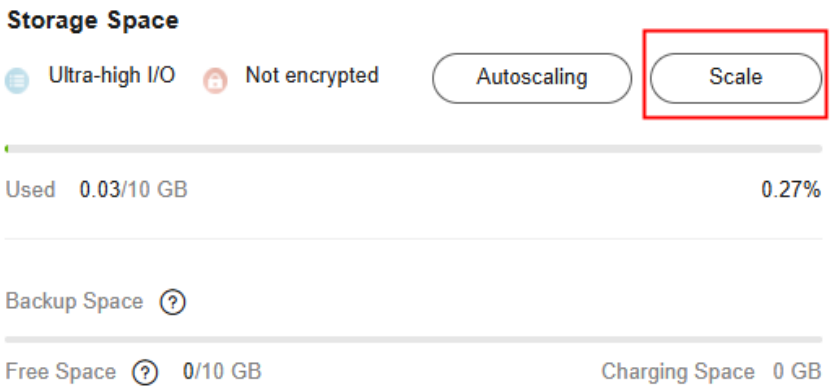
Step 4 On the **Instances** page, locate the replica set instance and choose **More** > **Scale Storage Space** in the **Operation** column.

Figure 7-3 Scale Storage Space



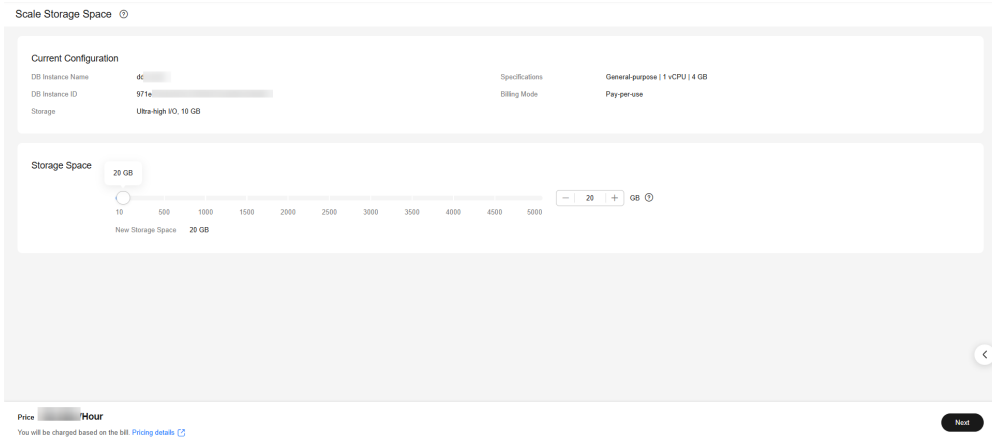
Alternatively, on the **Instances** page, click the name of the replica set instance. In the **Storage Space** area on the **Basic Information** page, click **Scale**.

Figure 7-4 Scale Storage Space



Step 5 On the displayed page, specify the desired amount of space to be changed and click **Next**.

Figure 7-5 Scale Storage Space



Select at least 10 GB each time you scale up the storage, and the storage size must be multiples of 10 GB. If a replica set instance has fewer than 8 vCPUs, the maximum storage that can be scaled up to is 5000 GB. If a replica set instance has 8 or more vCPUs, the maximum storage that can be scaled up to is 10,000 GB.

Step 6 On the displayed page, confirm the storage space.

- For yearly/monthly DB instances
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.
  - If you do not need to modify your settings, click **Submit** to go to the payment page and complete the payment.
- For pay-per-use DB instances
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.

  - If you do not need to modify the specifications, click **Submit** to scale up the storage space.

Step 7 Check the results.

- This process takes about 3 to 5 minutes. The status of the DB instance in the instance list is **Scaling up**.
- In the upper right corner of the instance list, click to refresh the list. The instance status changes to **Available**.
- In the **Storage Space** area on the **Basic Information** page, check whether the scaling up is successful.

CAUTION

If you need to scale up the storage space to more than 10 TB, submit a service ticket.

If the storage space is scaled up to more than 10 TB, the following risks may occur:

- If the purchased storage space exceeds 16,000 GB and the available storage space is 480 GB, the instance becomes read-only. After the available storage space is greater than 2,400 GB, the instance becomes readable and writable.
- If there is a large amount of data, the backup task may take a long time or even fail. In this case, the service SLA may be affected. You need to enable snapshot backup to ensure that the backup task can be executed properly. For details about how to enable snapshot backup, see **Enabling or Modifying an Automated Backup Policy**.
- If data is deleted by mistake, it takes a long time to restore a table to a specified point in time or restore a backup to a new instance, affecting the restoration efficiency.
- If the primary/secondary or read-only replication is delayed, it takes a long time to reconnect. As a result, the instance may be disconnected or fail to be reconnected.

----End

Reference

What Should I Do If Storage Usage Is Unusually High?

7.5.3 Scaling Up a Read Replica

Scenarios

If there is not enough storage for your workloads, you can scale up the storage of your DB instance.

- If an instance is created before September 2023, about 5% of the disk space is reserved for the system **root** user.
- If the storage usage reaches 97% of the purchased storage space, the database is set to the read-only state.

This section describes how to scale up the storage space of a read replica of a replica set instance.

Precautions

- Scaling is available when your account balance is sufficient.
- If you scale up a DB instance with disks encrypted, the expanded storage space will be encrypted using the original encryption key.
- An instance cannot be scaled up if it is in any of the following statuses:
  - Creating
  - Changing instance class
  - Adding node
  - Deleting node
  - Upgrading minor version
  - Switchover in progress
- During scaling, services will not be interrupted, and the storage type cannot be changed.

Billing

- A pay-per-use instance is still billed on an hourly basis after the instance is scaled up.
- If you scale up a yearly/monthly instance, you will pay price difference or get a refund.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the replica set instance name.

Step 5 In the **Node Information** area on the **Basic Information** page, locate the read replica you want to scale up and click **Scale Storage Space** in the **Operation** column.

Step 6 On the displayed page, specify the desired amount of space to be changed and click **Next**.

Select at least 10 GB each time you scale up the storage, and the storage size must be multiples of 10 GB. If a replica set instance has fewer than 8 vCPUs, the maximum storage that can be scaled up to is 5000 GB. If a replica set instance has 8 or more vCPUs, the maximum storage that can be scaled up to is 10,000 GB.

Step 7 On the displayed page, confirm the storage space.

- For yearly/monthly instances
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.

  - If you do not need to modify your settings, click **Submit** to go to the payment page and complete the payment.
- For pay-per-use instances
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.
  - If you do not need to modify the specifications, click **Submit** to scale up the storage space.

Step 8 Check the results.

- This process takes about 3 to 5 minutes. The status of the DB instance in the instance list is **Scaling up**.
- In the upper right corner of the instance list, click  to refresh the list. The instance status changes to **Available**.

CAUTION

If you need to scale up the storage space to more than 10 TB, submit a service ticket.

If the storage space is scaled up to more than 10 TB, the following risks may occur:

- If the purchased storage space exceeds 16,000 GB and the available storage space is 480 GB, the instance becomes read-only.
- If there is a large amount of data, the backup task may take a long time or even fail. In this case, the service SLA may be affected. You need to enable snapshot backup to ensure that the backup task can be executed properly. For details about how to enable snapshot backup, see **Enabling or Modifying an Automated Backup Policy**.
- If data is deleted by mistake, it takes a long time to restore a table to a specified point in time or restore a backup to a new instance, affecting the restoration efficiency.
- If the primary/secondary or read-only replication is delayed, it takes a long time to reconnect. As a result, the instance may be disconnected or fail to be reconnected.

----End

Reference

What Should I Do If Storage Usage Is Unusually High?

7.5.4 Scaling Up a Single Node Instance

Scenarios

If there is not enough storage for your workloads, you can scale up the storage of your DB instance. If you scale up the storage of a DB instance, the backup space increases accordingly.

- If an instance is created before September 2023, about 5% of the disk space is reserved for the system **root** user.
- If the storage usage reaches 97% of the purchased storage space, the database is set to the read-only state. After the storage usage drops below 85%, the instance becomes readable and writable.

In addition, you can set alarm rules for the storage usage. For details, see **Configuring Alarm Rules**.

For details about the causes and solutions of insufficient storage space, see **High Storage Usage**.

Precautions

- Scaling is available when your account balance is sufficient.
- If you scale up a DB instance with disks encrypted, the expanded storage space will be encrypted using the original encryption key.
- An instance cannot be scaled up if it is in any of the following statuses:
  - Creating
  - Changing instance class
  - Deleting node
  - Upgrading minor version
- Services are not interrupted during scaling. The storage type cannot be changed.

Billing

- A pay-per-use instance is still billed on an hourly basis after the instance is scaled up.
- If you scale up a yearly/monthly instance, you will pay price difference or get a refund.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the single node instance and choose **More** > **Scale Storage Space** in the **Operation** column.

Alternatively, on the **Instances** page, click the name of the single node instance. In the **Storage Space** area on the **Basic Information** page, click **Scale Storage Space**.

Step 5 On the displayed page, specify the desired amount of space to be changed and click **Next**.

Select at least 10 GB each time you scale up the storage, and the storage size must be multiples of 10 GB. The maximum amount of storage space is 1,000 GB.

Step 6 On the displayed page, confirm the storage space.

- For yearly/monthly DB instances
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.
  - If you do not need to modify your settings, click **Submit** to go to the payment page and complete the payment.
- For pay-per-use DB instances
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.
  - If you do not need to modify the specifications, click **Submit** to scale up the storage space.

Step 7 Check the results.

- This process takes about 3 to 5 minutes. The status of the DB instance in the instance list is **Scaling up**.
- In the upper right corner of the instance list, click to refresh the list. The instance status changes to **Available**.
- In the **Storage Space** area on the **Basic Information** page, check whether the scaling up is successful.

----End

Reference

What Should I Do If Storage Usage Is Unusually High?

7.6 Changing an Instance Class

7.6.1 Changing a Cluster Instance Class

Scenarios

This section describes how to change the class of a cluster instance.

Change Rules

Considering the stability and performance of DDS DB instances, you can change the DB instance class according to the rules listed in **Table 7-1**. Exercise caution when performing this operation.

Table 7-1 Change rules

Original Specification	Target Specification	Supported
General-purpose	General-purpose	√
	Enhanced	×

Original Specification	Target Specification	Supported
	Enhanced II	√
Enhanced	General-purpose	√
	Enhanced	×
	Enhanced II	√
Enhanced II	General-purpose	×
	Enhanced	×
	Enhanced II	√

NOTE

√ indicates that an item is supported, and × indicates that an item is not supported.

Precautions

- An instance cannot be deleted while its instance class is being changed.
- When the instance class is being changed, a primary/secondary switchover may occur once or twice and the database connection will be interrupted each time for up to 30s. You are advised to change the class during off-peak hours to reduce impacts and ensure that the service system of your client can reconnect to the database if the connection is interrupted.
- After the class of a cluster instance is changed, the system will change the value of **net.maxIncomingConnections** accordingly.
- A maximum of 16 shard nodes can be selected in each batch of class change.
- When the CPU or memory of the shard, config, or dds mongos node in a cluster instance is changed, the read replica class is not changed.
- The classes of read replicas in a cluster instance cannot be changed.
- The classes of yearly/monthly instance shard nodes can only be upgraded or downgraded one at a time.
- Changing the class does not cause data loss.
- If you forcibly change the class of an abnormal node in a DB instance, services may be interrupted.

Pre-check Items for Instance Class Change

- The instance status and the status of the node whose class is to be changed are normal.
- The primary/secondary replication delay does not exceed 20s. (This pre-check item applies only to shard and config nodes.)

Billing

- Instances billed on a pay-per-use basis are still billed based on the time used after the instance class is changed.
- If you change the class of a yearly/monthly instance, you will either pay for the difference or receive a refund.
  - If the price of the new instance class is higher than that of the original instance class, you need to pay for the price difference based on the used resource period.
  - If the price of the new instance class is lower than that of the original instance class, you will be refunded the difference based on the used resource period. The refund will be sent to your account. You can click **Billing Center** in the upper right corner of the console to view your account balance.

Changing dds mongos Class

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

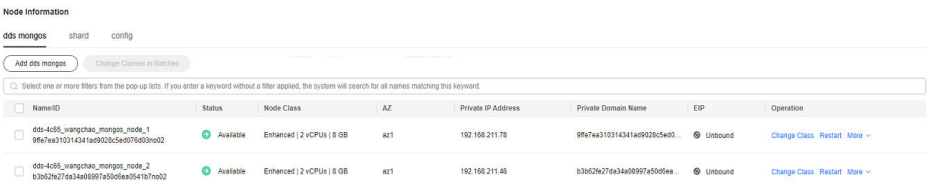
Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the cluster instance name.

Step 5 In the **Node Information** area on the **Basic Information** page, click the **dds mongos** tab. You can **change the class of a single dds mongos node** or **change the classes of multiple dds mongos nodes at a time**.

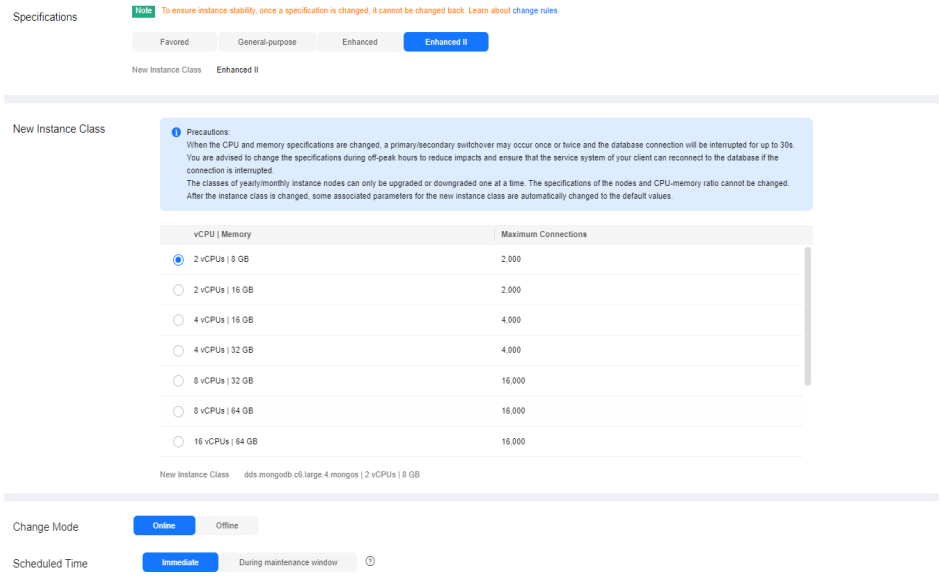
- Changing the class of a dds mongos node
  a. In the **Operation** column of the dds mongos node, click **Change Class**.

  Figure 7-6 Changing the class of a dds mongos node

  b. On the displayed page, select the required specifications, new class, change mode, and scheduled time, and click **Next**.

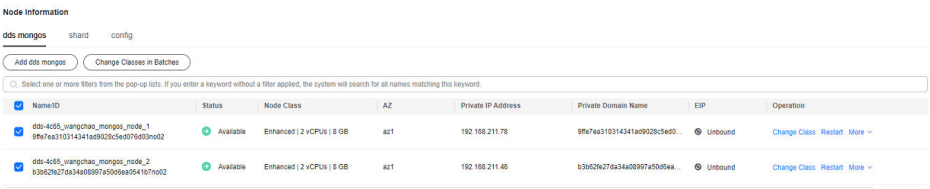
Figure 7-7 Changing dds mongos class



- Changing the classes of multiple dds mongos nodes in batches

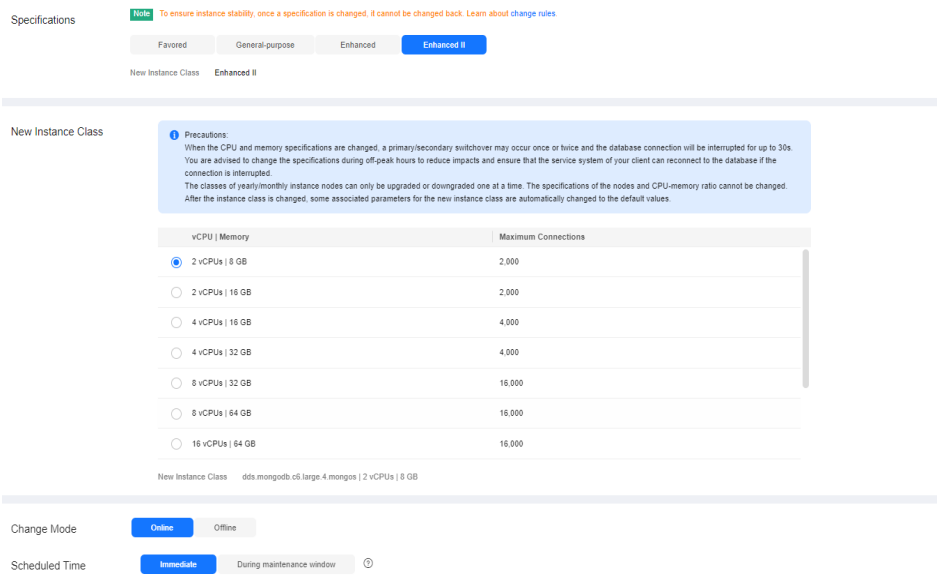
  a. Select the target dds mongos nodes and click **Change Classes in Batches**.

  Figure 7-8 Changing the classes of multiple dds mongos nodes in batches

  b. On the displayed page, select the required specifications, new class, change mode, and scheduled time, and click **Next**.

Figure 7-9 Changing the classes of multiple dds mongos nodes in batches



NOTE

- Online change: The specifications of multiple dds mongos nodes will be changed one by one. The time required depends on the number of instance nodes whose specifications need to be changed. Each node takes about 5 to 10 minutes. You are advised to connect to a DB instance using the HA connection address and ensure that your applications support automatic reconnection.
- Offline change: The specifications of multiple dds mongos nodes will be changed concurrently and the database will be unavailable during the specification change. It takes about 5 to 10 minutes. This function is now in OBT. To use it, submit a service ticket.
- The specifications change of dds mongos nodes does not involve primary/secondary switchovers.

Step 6 On the displayed page, confirm the class.

- If you need to modify your settings, click **Previous**.
- For pay-per-use instances

  If you do not need to modify your settings, click **Submit** to change the class. After the specifications are changed, you are still charged on an hourly basis.
- For yearly/monthly instances
  - If you intend to scale down the class, click **Submit**. The refund is automatically returned to your account.
  - If you intend to scale up the class, click **Pay Now**. The scaling starts only after the payment is successful.

Step 7 View the results.

- When the instance class is being changed, the status displayed in the **Status** column is **Changing instance class**. This process takes about 10 minutes.

- In the upper right corner of the instance list, click to refresh the list. The instance status changes to **Available**.
- In the **Node Information** area on the **Basic Information** page, click the **dds mongos** tab and view the new class.

----End

Changing shard Class

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

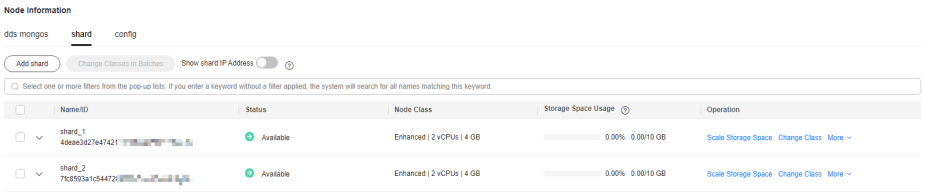
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the cluster instance name.

Step 5 In the **Node Information** area on the **Basic Information** page, click the **shard** tab. You can **change the class of a single shard** or **change the classes of multiple shards at a time**.

- Changing the class of a shard
  a. In the **Operation** column of the shard node, click **Change Class**.

  Figure 7-10 Changing the class of a shard node

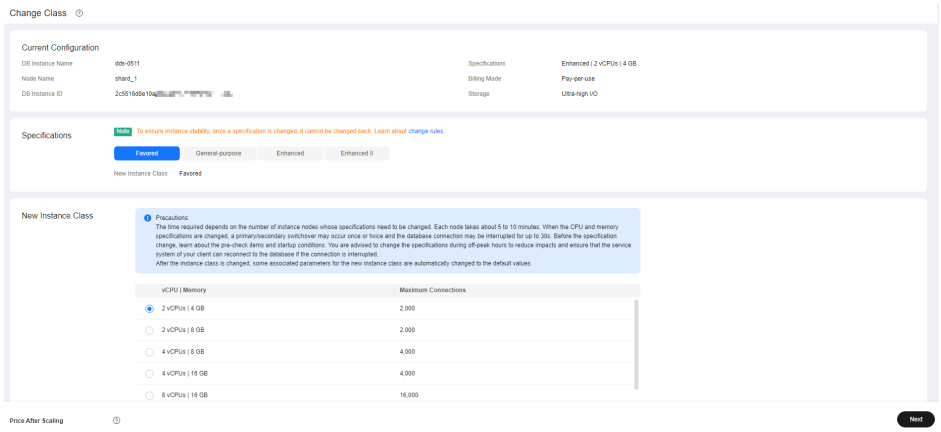
  

  b. On the displayed page, select the required specifications and new class and click **Next**.

  NOTE

  The time required depends on the number of instance nodes whose class is to be changed. It takes about 5 to 10 minutes for each node. When the instance class is being changed, a primary/secondary switchover may occur once or twice and the database connection will be interrupted each time for up to 30s. Before the class change, learn about **Pre-check Items for Instance Class Change**. You are advised to change the class during off-peak hours to reduce impacts and ensure that the service system of your client can reconnect to the database if the connection is interrupted.

Figure 7-11 Changing the class of a shard node

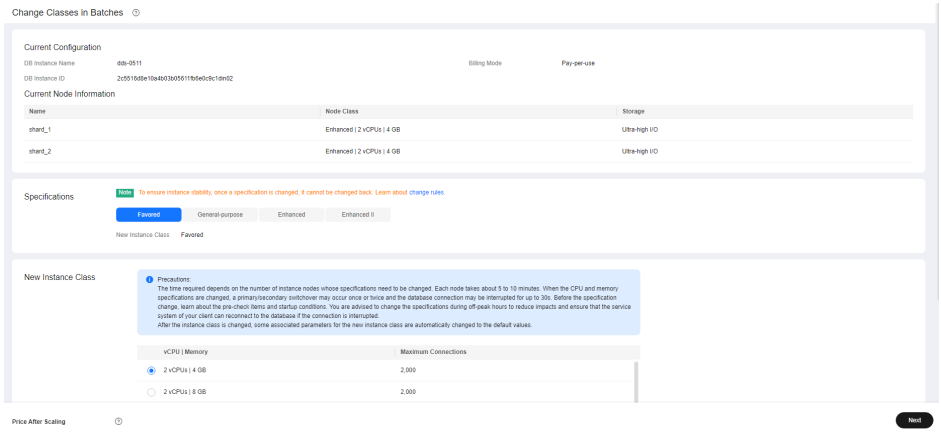


- Changing the classes of multiple shards in batches
  a. Select the target shards and click **Change Classes in Batches**.
  b. On the displayed page, select the required specifications and new class and click **Next**.

  NOTE

  The time required depends on the number of instance nodes whose class is to be changed. It takes about 5 to 10 minutes for each node. When the instance class is being changed, a primary/secondary switchover may occur once or twice and the database connection will be interrupted each time for up to 30s. Before the class change, learn about **Pre-check Items for Instance Class Change**. You are advised to change the class during off-peak hours to reduce impacts and ensure that the service system of your client can reconnect to the database if the connection is interrupted. This function is now in OBT. To use it, submit a service ticket.

Figure 7-12 Changing the classes of multiple shards in batches



Step 6 On the displayed page, confirm the class.

- If you need to modify your settings, click **Previous**.
- For pay-per-use instances

  If you do not need to modify your settings, click **Submit** to change the class. After the class is changed, you are still charged on an hourly basis.
- For yearly/monthly instances

  - If you intend to scale down the class, click **Submit**. The refund is automatically returned to your account.
  - If you intend to scale up the class, click **Pay Now**. The scaling starts only after the payment is successful.

Step 7 View the results.

- When the instance class is being changed, the status displayed in the **Status** column is **Changing instance class**. This process takes about 25 to 30 minutes.

  NOTE

  High database load increases the specification change duration. You are advised to change the specifications during off-peak hours to reduce impacts.

- In the upper right corner of the instance list, click to refresh the list. The instance status changes to **Available**.
- Go to the **Basic Information** page of the cluster instance you scaled up, click the **shard** tab in the **Node Information** area, and view the new class.

----End

Changing config Class

Step 1 Log in to the management console.

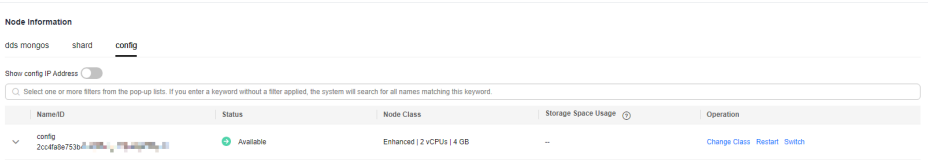
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the cluster instance name.

Step 5 In the **Node Information** area on the **Basic Information** page, click the **config** tab, locate the config node, and click **Change Class** in the **Operation** column.

Figure 7-13 Changing config class

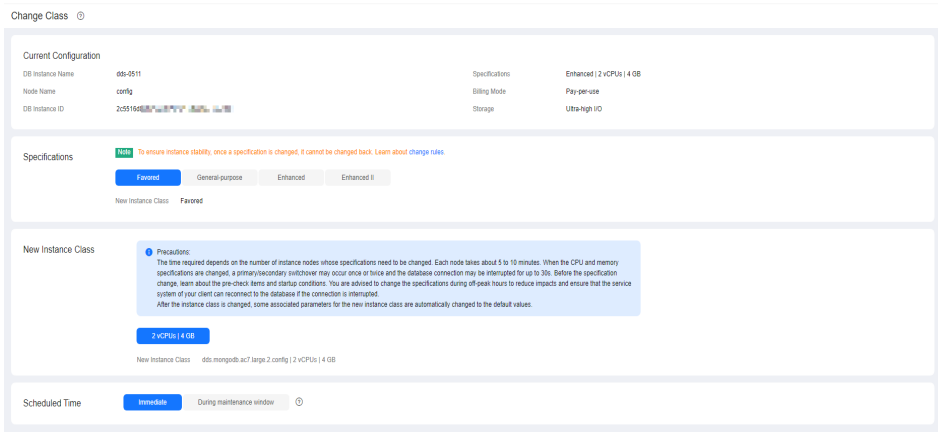


Step 6 On the displayed page, select the required specifications and new class and click **Next**.

NOTE

The time required depends on the number of instance nodes whose class is to be changed. It takes about 5 to 10 minutes for each node. When the instance class is being changed, a primary/secondary switchover may occur once or twice and the database connection will be interrupted each time for up to 30s. Before the class change, learn about **Pre-check Items for Instance Class Change**. You are advised to change the class during off-peak hours to reduce impacts and ensure that the service system of your client can reconnect to the database if the connection is interrupted. This function is now in OBT. To use it, submit a service ticket.

Figure 7-14 Changing config class



Step 7 On the displayed page, confirm the class.

- If you need to modify your settings, click **Previous**.
- For pay-per-use instances

  If you do not need to modify your settings, click **Submit** to change the class. After the class is changed, you are still charged on an hourly basis.
- For yearly/monthly instances
  - If you intend to scale down the class, click **Submit**. The refund is automatically returned to your account.
  - If you intend to scale up the class, click **Pay Now**. The scaling starts only after the payment is successful.

Step 8 View the results.

- When the instance class is being changed, the status displayed in the **Status** column is **Changing instance class**. This process takes about 25 to 30 minutes.

  NOTE

  High database load increases the specification change duration. You are advised to change the specifications during off-peak hours to reduce impacts.
- In the upper right corner of the instance list, click to refresh the list. The instance status changes to **Available**.
- Go to the **Basic Information** page of the cluster instance you scaled up, click the **config** tab in the **Node Information** area, and view the new class.

----End

Reference

How Do I Solve the High CPU Usage Issue?

7.6.2 Changing a Replica Set Instance Class

Scenarios

This section describes how to change the class of a replica set instance.

Change Rules

Considering the stability and performance of DDS DB instances, you can change the DB instance class according to the rules listed in **Table 7-2**. Exercise caution when performing this operation.

Table 7-2 Change rules

Original Specification	Target Specification	Supported
General-purpose	General-purpose	√
	Enhanced	×
	Enhanced II	√
Enhanced	General-purpose	√
	Enhanced	×
	Enhanced II	√
Enhanced II	General-purpose	×
	Enhanced	×
	Enhanced II	√

NOTE

√ indicates that an item is supported, and × indicates that an item is not supported.

Precautions

- An instance cannot be deleted while its instance class is being changed.
- When the CPU or memory of a replica set instance is changed, the read replica class is not changed.
- When the instance class is being changed, a primary/secondary switchover may occur once or twice and the database connection will be interrupted each time for up to 30s. You are advised to change the class during off-peak hours to reduce impacts and ensure that the service system of your client can reconnect to the database if the connection is interrupted.
- After the class of a replica set instance is changed, the system will change the value of **net.maxIncomingConnections** accordingly.
- Changing the class does not cause data loss.
- If you forcibly change the class of an abnormal node in a DB instance, services may be interrupted.

Pre-check Items for Instance Class Change

- The instance status and the status of the node whose class is to be changed are normal.

- The primary/secondary replication delay does not exceed 20s.

Billing

- Instances billed on a pay-per-use basis are still billed based on the time used after the instance class is changed.
- If you change the class of a yearly/monthly instance, you will either pay for the difference or receive a refund.
  - If the price of the new instance class is higher than that of the original instance class, you need to pay for the price difference based on the used resource period.
  - If the price of the new instance class is lower than that of the original instance class, you will be refunded the difference based on the used resource period. The refund will be sent to your account. You can click **Billing Center** in the upper right corner of the console to view your account balance.

Changing the Class of a Replica Set Instance

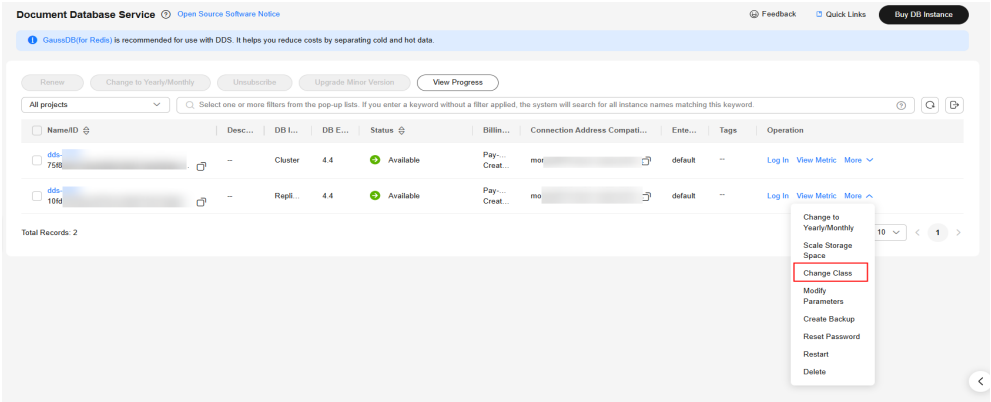
Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

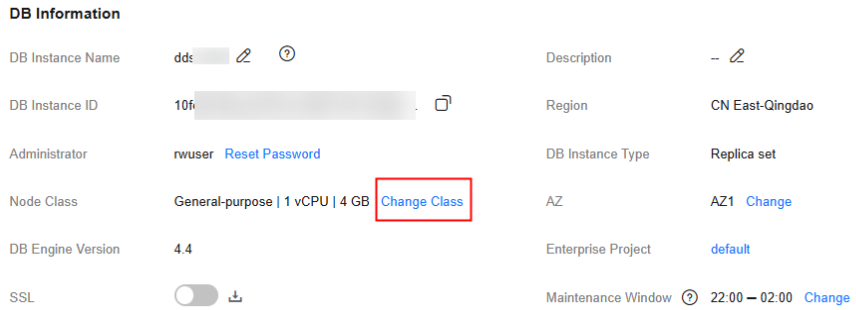
Step 4 On the **Instances** page, locate the replica set instance and choose **More** > **Change Instance Class** in the **Operation** column.

Figure 7-15 Changing the class of a replica set instance



Alternatively, on the **Instances** page, click the name of the replica set instance. In the **DB Information** area on the **Basic Information** page, click **Change Class** to the right of the **Node Class** field.

Figure 7-16 Changing the class of a replica set instance

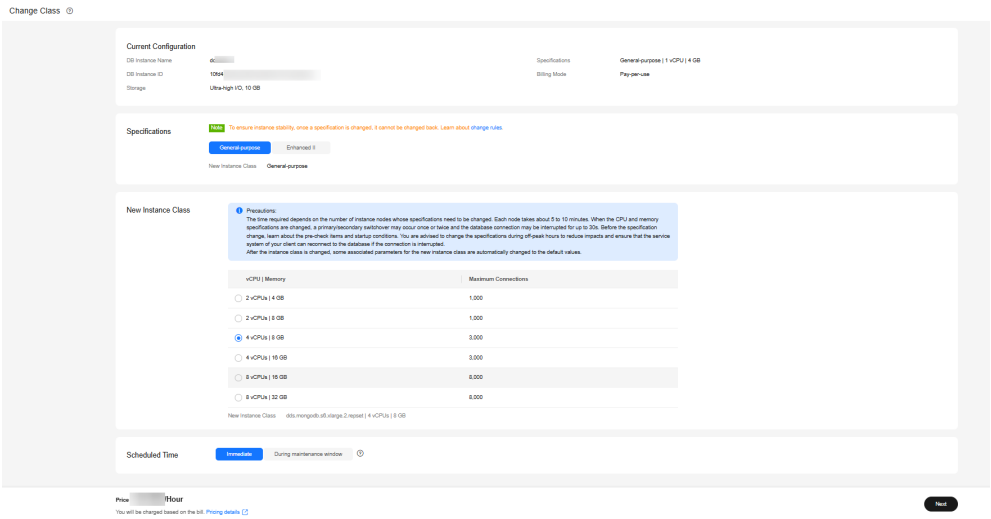


Step 5 On the displayed page, select the required specifications and new class and click **Next**.

NOTE

The time required depends on the number of instance nodes whose class is to be changed. It takes about 5 to 10 minutes for each node. When the instance class is being changed, a primary/secondary switchover may occur once or twice and the database connection will be interrupted each time for up to 30s. Before the class change, learn about **Pre-check Items for Instance Class Change**. You are advised to change the class during off-peak hours to reduce impacts and ensure that the service system of your client can reconnect to the database if the connection is interrupted.

Figure 7-17 Changing the class of a replica set instance



Step 6 On the displayed page, confirm the instance class.

- If you need to modify your settings, click **Previous**.
- For pay-per-use instances

  If you do not need to modify your settings, click **Submit** to change the instance class. After the class is changed, you are still charged on an hourly basis.
- For yearly/monthly instances
  - If you intend to scale down the instance class, click **Submit**. The refund is automatically returned to your account.

- If you intend to scale up the DB instance class, click **Pay Now**. The scaling starts only after the payment is successful.

Step 7 View the results.

- When the instance class is being changed, the status displayed in the **Status** column is **Changing instance class**. This process takes about 25 to 30 minutes.

  NOTE

  High database load increases the specification change duration. You are advised to change the specifications during off-peak hours to reduce impacts.

- In the upper right corner of the instance list, click to refresh the list. The instance status changes to **Available**.
- Go to the **Basic Information** page of the replica set instance you scaled up and check whether the scaling up is successful in the **DB Information** area.

----End

Changing the Class of a Read Replica

Step 1 Log in to the management console.

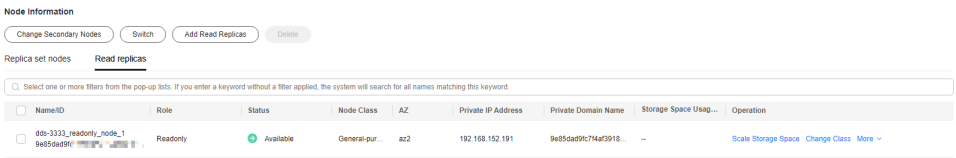
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the target replica set instance and click its name.

Step 5 In the **Node Information** area on the **Basic Information** page, click the **Read replicas** tab. Locate the read replica whose class you want to change, and click **Change Instance Class** in the **Operation** column.

Figure 7-18 Changing the class of a read replica

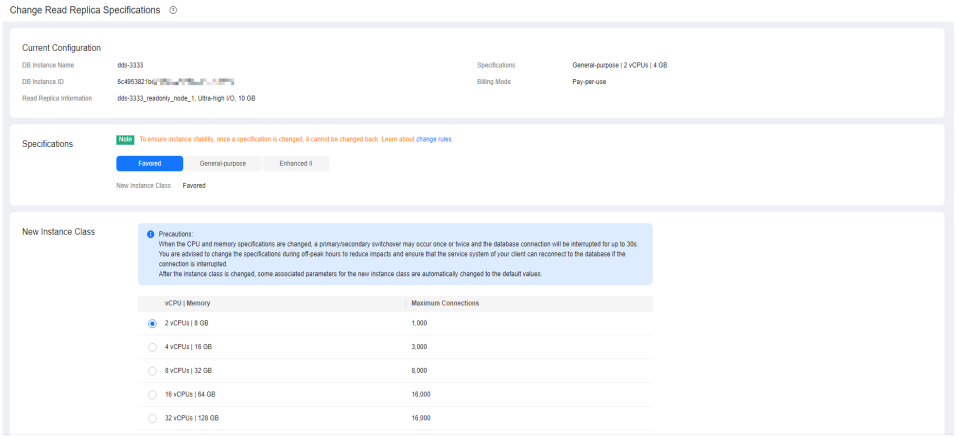


Step 6 On the displayed page, select the required specifications and new class and click **Next**.

NOTE

When the class of a read replica is being changed, there is a possibility that database access requests using the read replica fail. Before the class change, learn about **Pre-check Items for Instance Class Change**. You are advised to change the class during off-peak hours to reduce impacts and ensure that the service system of your client can reconnect to the database if the connection is interrupted.

Figure 7-19 Changing the class of a read replica



Step 7 On the displayed page, confirm the class.

- If you need to modify your settings, click **Previous**.
- For pay-per-use instances

  If you do not need to modify your settings, click **Submit** to change the class. After the class is changed, you are still charged on an hourly basis.
- For yearly/monthly instances
  - If you intend to scale down the class, click **Submit**. The refund is automatically returned to your account.
  - If you intend to scale up the class, click **Pay Now**. The scaling starts only after the payment is successful.

Step 8 View the results.

- When the class is being changed, the status displayed in the **Status** column is **Changing instance class**. This process takes about 25 to 30 minutes.
- In the upper right corner of the instance list, click to refresh the list. The instance status changes to **Available**.
- In the **Node Information** area on the **Basic Information** page, click the **Read replicas** tab. Locate the target read replica to view the new class.

----End

Reference

How Do I Solve the High CPU Usage Issue?

7.6.3 Changing a Single Node Instance Class

Scenarios

This section describes how to change the class of your single node instance.

Change Rules

Considering the stability and performance of DDS DB instances, you can change the DB instance class according to the rules listed in **Table 7-3**. Exercise caution when performing this operation.

Table 7-3 Change rules

Original Specification	Target Specification	Supported
General-purpose	General-purpose	√
	Enhanced	×
	Enhanced II	√
Enhanced	General-purpose	√
	Enhanced	×
	Enhanced II	√
Enhanced II	General-purpose	×
	Enhanced	×
	Enhanced II	√

NOTE

√ indicates that an item is supported, and × indicates that an item is not supported.

Precautions

- An instance cannot be deleted while its instance class is being changed.
- When the instance class is being changed, the database connection will be interrupted for 5 to 10 minutes. You are advised to change the class during off-peak hours to reduce impacts and ensure that the service system of your client can reconnect to the database if the connection is interrupted. After the restart is complete, the cached memory will be automatically cleared. The instance needs to be warmed up to prevent congestion during peak hours.
- After the class of a single node instance is changed, the system will change the value of **net.maxIncomingConnections** accordingly.
- Changing the class does not cause data loss.
- If you forcibly change the class of an abnormal node in a DB instance, services may be interrupted.

Pre-check Items for Instance Class Change

- The DB instance is in the **Available** status.

Billing

- Instances billed on a pay-per-use basis are still billed based on the time used after the instance class is changed.
- If you change the class of a yearly/monthly instance, you will either pay for the difference or receive a refund.
  - If the price of the new instance class is higher than that of the original instance class, you need to pay for the price difference based on the used resource period.
  - If the price of the new instance class is lower than that of the original instance class, you will be refunded the difference based on the used resource period. The refund will be sent to your account. You can click **Billing Center** in the upper right corner of the console to view your account balance.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the single node instance and choose **More** > **Change Instance Class** in the **Operation** column.

Alternatively, on the **Instances** page, click the name of the single node instance. In the **DB Information** area on the **Basic Information** page, click **Change** to the right of the **Node Class** field.

Step 5 On the displayed page, select the required specifications and new class and click **Next**.

Step 6 On the displayed page, confirm the instance class.

- If you need to modify your settings, click **Previous**.
- For pay-per-use instances

  If you do not need to modify your settings, click **Submit** to change the instance class. After the class is changed, you are still charged on an hourly basis.
- For yearly/monthly instances
  - If you intend to scale down the instance class, click **Submit**. The refund is automatically returned to your account.
  - If you intend to scale up the instance class, click **Pay Now**. The scaling starts only after the payment is successful.

Step 7 View the results.

- When the instance class is being changed, the status displayed in the **Status** column is **Changing instance class**. This process takes about 10 minutes.

NOTE

High database load increases the specification change duration. You are advised to change the specifications during off-peak hours to reduce impacts.

- In the upper right corner of the instance list, click to refresh the list. The instance status changes to **Available**.
- Go to the **Basic Information** page of the single node you scaled up and check whether the scaling process is successful in the **Configuration** area.

----End

Reference

How Do I Solve the High CPU Usage Issue?

7.7 Changing Cluster Instance Nodes

7.7.1 Adding Cluster Instance Nodes

Scenarios

As service data increases, the number of current database nodes cannot meet the service requirements. In this case, you can add more nodes to the instance.

Precautions

- To add nodes, instance status must be **Available**, **Deleting backup**, or **Checking restoration**.
- Nodes cannot be added to a DB instance that is being backed up.
- A DB instance cannot be deleted while nodes are being added.
- An instance node can be added within 5 minutes. The time required depends on the number of nodes to be added.
- Adding nodes does not affect cluster services.
- When adding a shard node for a cluster DB instance, ensure that the node class is greater than or equal to the highest class of a shard in the instance.

Billing

- A pay-per-use instance is still billed on an hourly basis after new nodes are added.
- If you add nodes to a yearly/monthly instance, you will pay price difference or get a refund.

Adding dds mongos Nodes

Step 1 Log in to the management console.

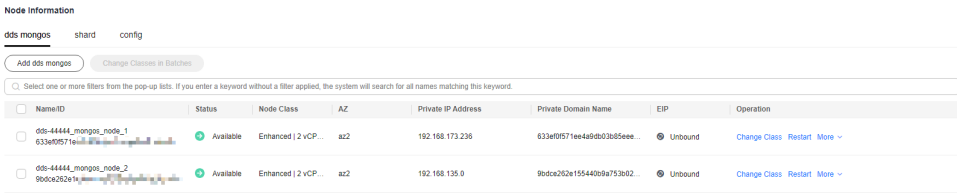
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click ☰ in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the cluster instance name.

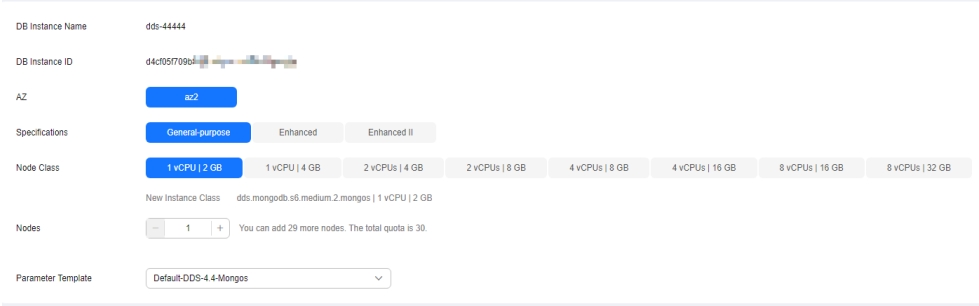
Step 5 On the **dds mongos** tab in the **Node Information** area, click **Add dds mongos**.

Figure 7-20 Node information



Step 6 On the displayed page, specify **Node Class**, **Nodes**, and **Parameter Template** and click **Next**.

Figure 7-21 Adding dds mongos nodes



A Community Edition cluster instance supports up to 32 dds mongos nodes.

Step 7 On the displayed page, confirm the node configuration information.

- Yearly/Monthly
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.
  - If you do not need to modify your settings, click **Submit** to go to the payment page and complete the payment.
- Pay-per-use
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.
  - If you do not need to modify your settings, click **Submit** to add the nodes.

Step 8 View the results.

- This process takes about 10 to 15 minutes. During that time, the status of the DB instance in the instance list is **Adding node**.
- In the upper right corner of the DB instance list, click ↻ to refresh the list. The instance status changes to **Available**.

- On the **dds mongos** tab in the **Node Information** area, view the information about the node you added.
- If the dds mongos nodes fail to be added, you can revert them in batches or delete them one by one. For details, see section **Reverting Cluster Instance Nodes**.

----End

Adding shard Nodes

Step 1 Log in to the management console.

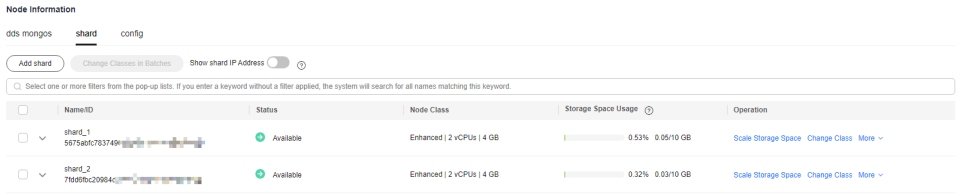
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instance Management** page, click the target cluster instance.

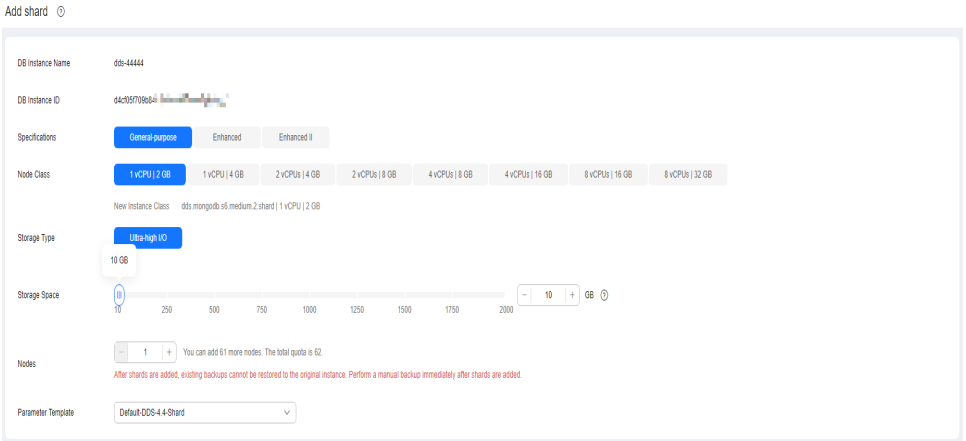
Step 5 On the **shard** tab in the **Node Information** area, click **Add shard**.

Figure 7-22 Node information



Step 6 Specify **Node Class**, **Storage Space**, **Nodes**, and **Parameter Template** and click **Next**.

Figure 7-23 Adding shard nodes



- The storage space you applied for will include the system overhead required for inode, reserved block, and database operation. The storage space must be a multiple of 10.
- A Community Edition cluster instance supports up to 32 shard nodes.

Step 7 On the displayed page, confirm the node configuration information.

- Yearly/Monthly
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.
  - If you do not need to modify your settings, click **Submit** to go to the payment page and complete the payment.
- Pay-per-use
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.
  - If you do not need to modify your settings, click **Submit** to add the nodes.

Step 8 View the results.

- This process takes about 10 to 15 minutes. During that time, the status of the DB instance in the instance list is **Adding node**.
- In the upper right corner of the DB instance list, click  to refresh the list. The instance status changes to **Available**.
- On the **shard** tab in the **Node Information** area, view the information about the node you added.
- If shard addition fails, you can roll back the operation in batches or delete shards one by one. For details, see **Reverting Cluster Instance Nodes**.

----End

7.7.2 Reverting Cluster Instance Nodes

Scenarios

This section describes how to roll back a failed node addition.

Reverting Nodes in Batches

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the cluster instance to which nodes fail to be added and choose **More** > **Revert** in the **Operation** column.

Step 5 In the displayed dialog box, click **Yes**.

During the rollback, the instance status is **Deleting node**. This process takes about 1 to 3 minutes.

----End

Deleting a Single Node

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the cluster instance to which the node fails to be added.

Step 5 In the **Node Information** area on the **Basic Information** tab, click the **dds mongos** or **shard** tab, locate the dds mongos node, shard node, or read replica that fails to be added, and click **Delete**.

Step 6 In the displayed dialog box, click **Yes**.

During deletion, the node status is **Deleting node**. This process takes about 1 to 3 minutes.

----End

7.8 Changing Replica Set Instance Nodes

7.8.1 Changing Replica Set Instance Nodes

Scenarios

DDS allows you to add nodes to and delete nodes from a replica set instance.

DDS allows you to scale out a three-node replica set instance to up to five or even seven nodes. All newly added nodes are secondary nodes and support primary/secondary switchovers, improving data reliability. You can also delete unnecessary nodes to release resources.

Precautions

- To add nodes, instance status must be **Available**, **Deleting backup**, or **Checking restoration**.
- An instance cannot be deleted when nodes are being added.
- If there are any newly added secondary nodes, they will be unable to participate in this switchover. When you add a new secondary node, the HA connection address needs to be reconfigured, and the new node is frozen for 12 hours.
- When instance nodes are being added, the DB instance may be intermittently disconnected once or twice for up to 30s each time.
- Deleted nodes cannot be recovered. Exercise caution when performing this operation.
- If you enable operation protection, two-factor authentication is required for sensitive operations to secure your account and cloud products. For details about how to enable operation protection, see .

- Nodes cannot be deleted from instances that have abnormal nodes.

Billing

- A pay-per-use instance is still billed on an hourly basis after new nodes are added.
- If you change nodes of a yearly/monthly instance, you will pay price difference or get a refund.

Adding Replica Set Instance Nodes

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

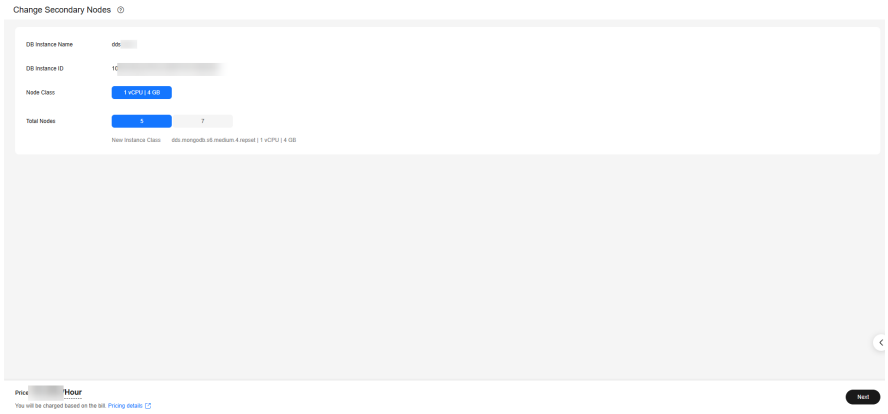
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the replica set instance name.

Step 5 In the **Node Information** area on the **Basic Information** page, click **Change Secondary Nodes**.

Step 6 Specify **Total Nodes** and click **Next**.

Figure 7-24 Selecting the number of nodes



You can add nodes to a five-node or seven-node replica set as required.

Step 7 On the displayed page, confirm the node configuration information.

- For yearly/monthly DB instances
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.
  - If you do not need to modify your settings, click **Submit** to go to the payment page and complete the payment.
- For pay-per-use DB instances
  - If you need to modify your settings, click **Previous** to go back to the page for you to specify details.

  - If you do not need to modify your settings, click **Submit** to add nodes.

Step 8 Check the results.

- When nodes are being added, the status of the instance is **Adding node**. The entire process takes about 15 minutes.
- In the **Node Information** area, you can see the information about the nodes you added.

----End

Deleting Replica Set Instance Nodes

Step 1 Log in to the management console.

Step 2 In the service list, choose **Databases** > **Document Database Service**.

Step 3 On the **Instances** page, locate the target DB instance and click its name.

Step 4 In the **Node Information** area on the **Basic Information** page, click **Change Secondary Nodes**.

Step 5 Specify **Total Nodes** and click **Next**.

NOTE

When you delete nodes, the number of selected nodes should be less than the number of current nodes. For example, if the number of current instance nodes is 5, select **3** when deleting nodes.

Step 6 Click **Submit**.

Step 7 If you have enabled operation protection, click **Start Verification** in the **Delete Node** dialog box. On the displayed page, click **Send Code**, enter the verification code, and click **Verify**. The page is closed automatically.

----End

7.8.2 Adding Read Replicas to a Replica Set Instance

Scenarios

Read replicas enhance read capabilities and reduce load on primary nodes. After a DDS replica set instance is created, you can create read replicas based on service requirements. To connect to a read replica, see **Connecting to a Read Replica Using Mongo Shell**.

NOTE

This function is now in OBT. To use it, submit a service ticket.

Constraints

- To use this function, choose **Service Tickets** > **Create Service Ticket** to submit a service ticket.
- The version of a replica set instance must be 3.4, 4.0, 4.2, 4.4 or 5.0.
- Nodes cannot be added to an instance that is being backed up.

- An instance cannot be deleted when one or more nodes are being added.
- When read replicas are being added, the DB instance may be intermittently disconnected once or twice for up to 30s each time.

Precautions

- A maximum of five read replicas can be added to a replica set instance.

Procedure

Step 1 Log in to the management console.

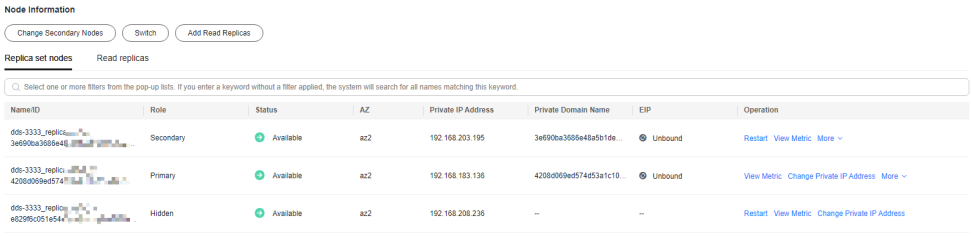
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the replica set instance.

Step 5 In the **Node Information** area on the **Basic Information** page, click **Add Read Replicas**.

Figure 7-25 Creating read replicas



Step 6 On the **Add Read Replicas** page, specify **Specifications**, **Node Class**, **Nodes**, **Parameter Template**, and **Delay**, and click **Next**.

Figure 7-26 Creating read replicas

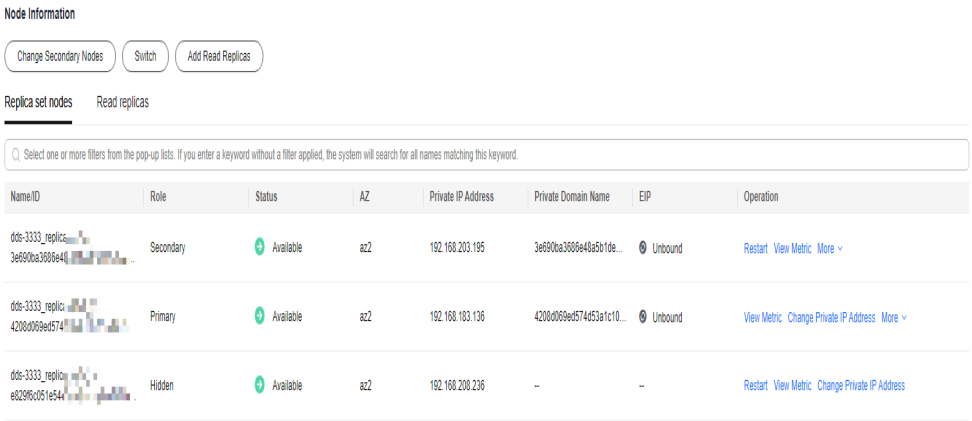


Table 7-4 Parameter description

Parameter	Description
Read Replica Parameter Template	The parameters that apply to the read replicas of a replica set instance. After a node is created, you can change the parameter template of the node for optimal performance.

Step 7 On the displayed page, confirm the node configuration information.

- If you need to modify your settings, click **Previous** to go back to the page for you to specify details.
- If you do not need to modify your settings, click **Submit** to add nodes.

Step 8 View the results.

- When nodes are added, the status of the instance is **Adding read replicas**. The entire process takes about 15 minutes.
- In the **Node Information** area, view the information about the nodes you added.
- Choose **More** > **View Delay** in the **Operation** column to view the delay of the current node.

----End

7.8.3 Manually Switching the Primary and Secondary Nodes of a Replica Set

Scenarios

A replica set consists of the primary node, secondary node, and hidden node. Primary and secondary nodes allow access from external services by providing IP addresses. Hidden nodes are only used for backing up data. When a primary node becomes faulty, the system automatically selects a new primary node to ensure high availability. DDS supports primary/secondary switchovers for scenarios such as disaster recovery.

Precautions

- To perform a switchover, the instance status needs to be **Available**, **Changing to yearly/monthly**, and **Changing a security group**.
- The database connection may be interrupted during the switchover. Ensure that your client supports reconnection.
- If there are any newly added secondary nodes, they will be unable to participate in this switchover. When you add a new secondary node, the HA connection address needs to be reconfigured, and the new node is frozen for 12 hours.
- A primary/secondary switchover can be performed only when the DB instance is available.
- The longer the delay for primary/secondary synchronization, the more time is needed for a primary/secondary switchover. If the primary to secondary synchronization delay exceeds 300s, primary/secondary switchover is not

supported. For details about the synchronization delay, see **What Is the Time Delay for Primary/Secondary Synchronization in a Replica Set?**

Performing a Primary/Secondary Switchover

Step 1 Log in to the management console.

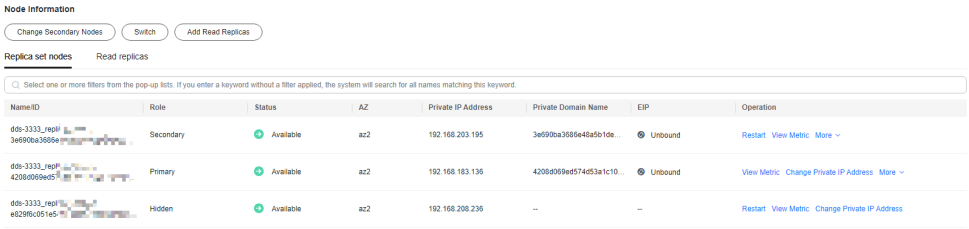
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the replica set instance.

Step 5 In the **Node Information** area on the **Basic Information** page, click **Switch**.

Figure 7-27 Primary/Secondary switchover



Step 6 In the displayed dialog box, click **Yes**.

Step 7 Check the result.

- During the switchover process, the DB instance status changes to **Switchover in progress**. After the switchover is complete, the status is restored to **Available**.
- In the **Node Information** area, you can view the switchover result.
- After the switchover, the previous primary node becomes the secondary node. You need to reconnect to the primary node. For details, see **Connecting to a DB Instance**.

----End

Forcibly Promoting a Secondary Node to the Primary

Step 1 Log in to the management console.

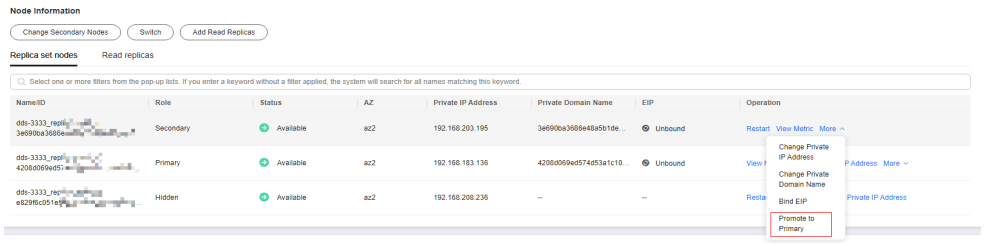
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the target replica set instance and click its name.

Step 5 In the **Node Information** area on the **Basic Information** page, locate a target node whose role is **Secondary** and click **Promote to Primary** in the **Operation** column.

Figure 7-28 Promote to Primary



Step 6 In the displayed dialog box, click **Yes**.

Step 7 Check the result.

- In the **Node Information** area on the **Basic Information** page, you can view the result.

----End

7.9 Configuring the Maintenance Window

During a maintenance window, Huawei Cloud O&M personnel perform maintenance operations on your instance. To prevent service interruptions, set the maintenance window to off-peak hours.

The maintenance window is 02:00–06:00 by default and you can change it as required.

Precautions

- Before maintenance is performed, DDS will send SMS and email notifications to the contact person you specified in the Huawei account.
- During the maintenance window, the DB instance may be intermittently disconnected once or twice. Ensure that your applications can reconnect to the database if the connection is interrupted.
- During DB instance maintenance, operations for service changes (such as upgrade and restart) are unavailable except account management, database management, and security group adding. When a DB instance is in maintenance, data access and query operations on the database are not affected.
- Changing the maintenance window does not affect the execution of tasks that have been scheduled.
- You can configure a maintenance window only for restarting a DB instance, changing an instance class, or upgrading the minor version of a DB instance.
- You can cancel a scheduled task to be executed on the **Task Center** page.
- The maintenance window cannot overlap the time window configured for backups. Otherwise, scheduled tasks may fail.
- Tasks delivered near the end of the maintenance window may fail to be scanned. In this case, the execution is canceled.

Changing a Maintenance Window

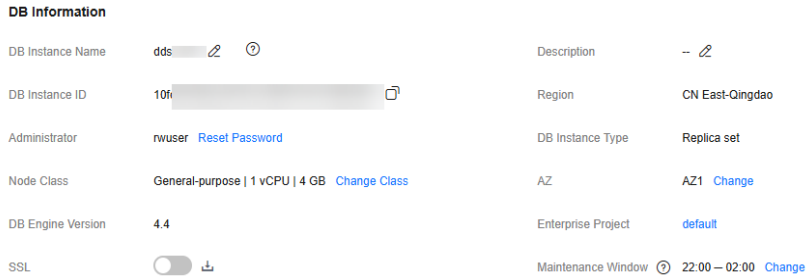
Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

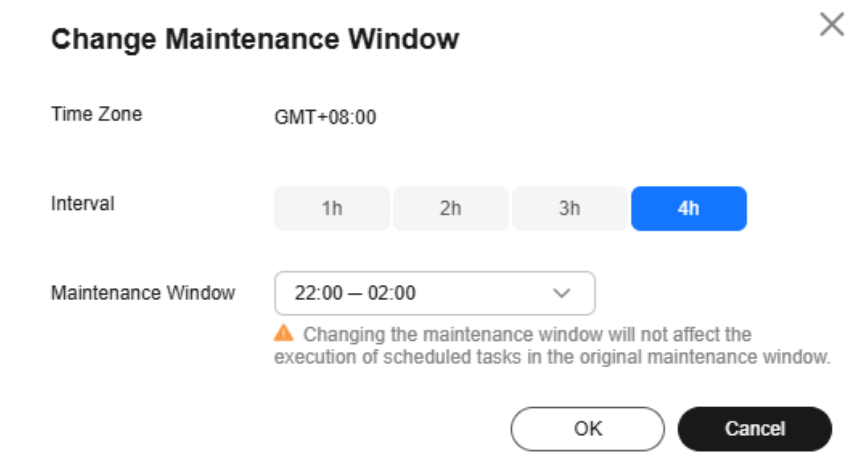
Step 4 On the **Instances** page, click the instance name. In the **DB Information** area on the **Basic Information** page, click **Change** in the **Maintenance Window** field.

Figure 7-29 Changing the maintenance window



Step 5 In the displayed dialog box, select an interval and a maintenance window, and click **OK**.

Figure 7-30 Changing the maintenance window



----End

Canceling a Scheduled Task

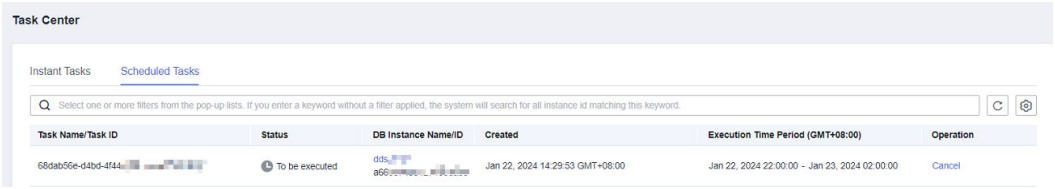
Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click ☰ in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Task Center** page, locate the specified task and click **Cancel** in the **Operation** column.

Figure 7-31 Canceling a task



Step 5 View the result.

On the **Task Center** page, you can view the result. After the task is cancelled, its status changes to **Cancelled**.

----End

7.10 Changing an AZ

Scenarios

You can migrate a DB instance to another AZ in the same region as the original AZ.

Precautions

- Clusters and replica sets can be migrated between AZs.
- Instances deployed across AZs and associated with an IPv6 subnet do not support this operation.
- Inactive secondary nodes and read replicas in a replica set instance do not support this operation.
- If a cluster instance has read replicas associated, the instance cannot be migrated to another AZ.
- Services will be interrupted for up to 60 seconds while the AZ is being changed. The time required to change an AZ depends on the amount of data to be migrated. You are advised to change an AZ during off-peak hours. You are advised to use an HA connection to access the instance or configure your client to automatically reconnect to the instance.
- The destination AZ and the AZ of the current DB instance are in the same region.
- For details about regions and AZs, see **Regions and AZs**.
- To ensure stable operation of a DB instance, change an AZ during off-peak hours.

Supported Migration Types and Scenarios

Table 7-5 Supported migration types and scenarios

Migration Type	Scenario
Migrating data from one AZ to another AZ	DDS instances can be migrated to the AZ to which the ECS belongs. DDS instances and ECS in the same AZ can be connected through a private network with lower network latency.
Migrating data from a single AZ to multiple AZs	The instance disaster recovery capability needs to be improved.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the **DB Information** area on the **Basic Information** page, click **Change** to the right of the **AZ** field.

Step 6 On the displayed page, select a desired AZ and click **OK**.

Step 7 On the **Instances** page, check the changed AZ.

- During the changes, the instance status is **Changing AZ**.
- In the upper right corner of the instance list, click  to refresh the list. After the migration is complete, the instance status will become **Available**.
- In the **DB Information** on the **Basic Information** page, view the new AZ where the DB instance is deployed.

----End

8 Data Backups

8.1 Backup Principles and Solutions

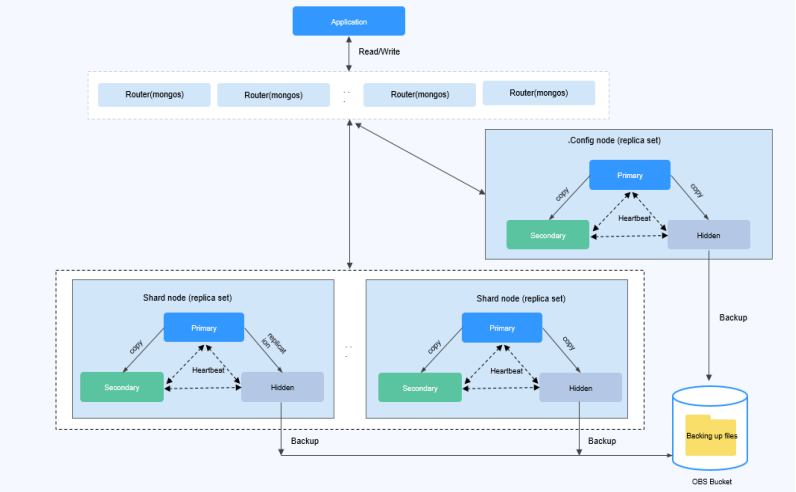
DDS instances support automated and manual backups. You can periodically back up databases. If a database is faulty or data is damaged, you can restore the database using backup files to ensure data reliability.

Backup Principles

- Cluster instance

  A cluster instance consists of a config node, and multiple dds mongos and shard nodes. The config node is used to store the configuration information of a cluster instance, and the shard node is used to store data of a cluster instance. Backing up a cluster instance means that data on the config and shard nodes is backed up separately. As shown in **Figure 8-1**, the config and shard nodes in a cluster instance are backed up to their own hidden nodes. The backup process occupies certain CPU and memory resources of the hidden nodes. During the backup, the CPU usage, memory usage, and primary/standby delay of the hidden node increase slightly, which is normal. The backup files on the hidden nodes will then be compressed and stored in OBS, and the storage space of the instance will not be occupied.

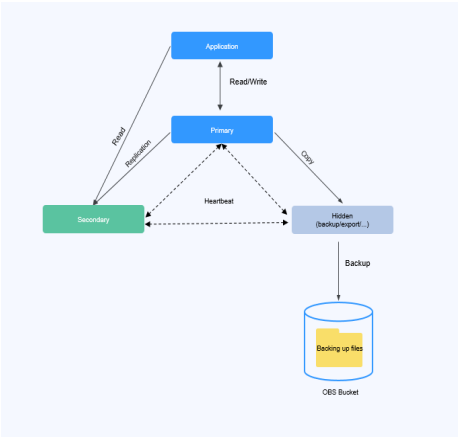
Figure 8-1 Cluster backup principle



- Replica set instance

  As shown in **Figure 8-2**, replica set instance data is backed up on hidden nodes. The backup process occupies certain CPU and memory resources of the hidden node. During the backup, the CPU usage, memory usage, and primary/standby delay of the hidden node increase slightly, which is normal. The backup files on the hidden nodes will then be compressed and stored in OBS, and the storage space of the instance will not be occupied.

Figure 8-2 Replica set backup principle



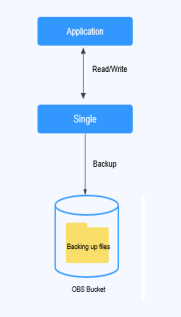
- Single node instance:

  Single-node instance backup is performed on only one node. The backup file is stored in OBS as a package, which does not occupy the storage of the instance.

NOTICE

A single node instance is backed up using mongodump. During the backup, CPU and memory resources of the node are occupied. If the resources are insufficient, the backup fails. You are advised to migrate the single-node instance data to a replica set instance for backup.

Figure 8-3 Single-node instance backup principle



Backup and Restoration Solution

- Table 8-1 describes how to back up and download backup files.

Table 8-1 Backup solutions

Task Type	Method	Supported Instance Type	Scenario
Backing up data	**Automated backup**	• Cluster • Replica set • Single node	You can perform automated backup for DDS instances on the management console.
	Incremental backup	• Cluster • Replica set	You can perform incremental backup for DDS instances on the management console.
	Manual backup	• Cluster • Replica set • Single node	You can perform manual backup for DDS instances on the management console.
	mongodump	• Cluster • Replica set • Single node	You can use the backup and restoration tool provided by the MongoDB client to back up your self-built MongoDB database or MongoDB database on the cloud.
	mongoexport	• Cluster • Replica set • Single node	You can use the backup and restoration tool provided by the MongoDB client to back up your self-built MongoDB database or MongoDB database on the cloud.
Downloading a backup file	**OBS Browser+**	• Cluster • Replica set • Single node	If the size of a backup file is greater than 400 MB, use OBS Browser+ to download the file.

Task Type	Method	Supported Instance Type	Scenario
	Browser	• Replica set • Single node	You can directly download backup files using a browser.
	URL	• Cluster • Replica set • Single node	You can download backup files in a new browser window, or using Xunlei or Wget.

- For details about the DDS restoration scheme, see **Solutions**.

Billing

Backups are saved as packages in OBS buckets. Backups occupy backup space in OBS. If the free space DDS provides is used up, the additional space required will be billed. For the billing details, see

8.2 Backup Types

DDS supports multiple backup types. Based on different dimensions, there are the following backup types:

Full Backups and Incremental Backups Based on Data Volume

Table 8-2 Comparison between full backups and incremental backups

Backup Type	Full backups	Incremental backups
Description	All data in an instance is backed up.	Only data changes within a certain period of time are backed up.
Enabled by Default	Yes	Yes
Retention Period	• You can specify how many days automated backups can be retained for. If you shorten the retention period, the new backup policy takes effect for existing backups. • Manual backups are always retained even though a DDS instance is deleted. They can only be deleted manually.	Incremental backups will be deleted along with automated full backups.

Characteristics	• A full backup is to back up all data of your DB instance in the current point of time. • You can use a full backup to restore the complete data generated when the backup was created. • Full backups can be created automatically or manually.	• Incremental backup is used to back up the data newly added or modified since the last full or incremental backup. • DDS automatically backs up the updated data every 5-60 minutes since the last automated or incremental backup was made.
How to Check Backup Size	Click the target instance name. On the **Backups & Restorations** page, click the **Instance-level Backups** tab and check the backup size.	Click the target instance name. On the **Backups & Restorations** page, click the **Incremental Backup** tab and check the backup size.

Automated Backups and Manual Backups Based on Backup Methods

Table 8-3

Backup Type	Automated backups	Manual backups
Description	• You can set an automated backup policy on the console, and the system will back up your instance data based on the time window and backup cycle you set in the backup policy and will store the backups for the retention period you specified. • Automated backups cannot be manually deleted. To delete them, you can adjust the retention period specified in your automated backup policy. Retained backups (including full and incremental backups) will be automatically deleted at the end of the retention period.	• Manual backups are user-initiated full backups of your DB instance. They are retained until you delete them manually. • Regularly backing up your DB instance is recommended, so if your DB instance fails or data is corrupted, you can restore it using backups.
Enabled by Default	Yes	Yes

Retention Period	Automated backups are retained for the number of days you specified. The retention period ranges from 1 to 732 days.	Manual backups are always retained until you delete them manually.
How to Configure	**Configuring an Automated Backup Policy**	**Creating a Manual Backup**

8.3 Performing Backups

8.3.1 Configuring an Automated Backup Policy

Scenarios

DDS backs up data automatically based on the automated backup policy you set. You are advised to regularly back up data in your database. If the database becomes faulty or data is damaged, you can restore it with the backup.

The automated backup policy for DDS is enabled by default. After an instance is created, you can **modify** or **disable** the automated backup policy as required.

Once the automated backup policy is enabled, a full backup is triggered immediately. After that, full backups will be created based on the backup window and backup cycle you specify. When an instance is being backed up, data is copied and then compressed and uploaded to OBS. The length of time the backup data is kept for depends on the backup retention period you configure. The backup duration depends on the amount of data, and the average backup speed is 60 MB/s. After the automated backup policy is enabled, an incremental backup is automatically performed every 5 minutes for replica set instances to ensure data reliability. If the incremental backup function is required for cluster instances, you need to manually enable it.

Automated Backup Description

- Backup type
  - Full backup: All data is backed up even if no data is updated since the last backup.
  - Incremental backup: Incremental backup is used to back up the data newly added or modified since the last automated full backup or incremental backup. DDS automatically backs up the updated data every 5 minutes since the last automated full backup or incremental backup was made.
- Backup mode
  - **Physical**: Data is copied from physical disks.
  - Snapshot

- The data status at a particular point in time is retained. Compared with physical backup, snapshot backup is faster. After CBR is enabled, the free backup space is unavailable. You are billed for database backup vaults on a pay-per-use basis. For details, see **How Is CBR Billed?**

  NOTE

  - The backup duration is proportional to how much data your instance has. Too much data can decrease the backup efficiency. If you have large amounts of data and want to speed up the backup process, submit a service ticket to enable CBR snapshot backup.
  - After CBR is enabled, snapshot backup is used. Existing automated and manual backups can still be used to restore data.
  - When you delete a DB instance, its automated backups are also deleted but its manual backups are retained.
  - After CBR is enabled, the next full backup is a snapshot backup. You can use the snapshot backup to restore data.

- If more than 2 TB of data needs to be backed up, the backup method cannot be set back to physical backup.
- Snapshots cannot be backed up across regions.

– **Logical**: A tool is used to read data and logically export the data.

- **Table 8-4** lists the automated backup methods supported by DDS.

Table 8-4 Backup methods

Instance Type	Backup Mode	Backup Type
Cluster	Physical backup/ Snapshot backup **NOTE** • This function is now in OBT. To use it, submit a service ticket. • For details about how to set **Backup Method** to **Snapshot**, see **Setting Backup Method for a DB Instance**.	• Full backup • Incremental backup **NOTE** Snapshot backup only applies to physical data in a full backup.

Instance Type	Backup Mode	Backup Type
Replica set	Physical backup/ Snapshot backup **NOTE** • This function is now in OBT. To use it, submit a service ticket. • For details about how to set **Backup Method** to **Snapshot**, see **Setting Backup Method for a DB Instance**.	• Full backup • Incremental backup **NOTE** Snapshot backup only applies to physical data in a full backup.
Single node **NOTE** Single node instances apply to only a few scenarios. You are advised to use a single node instance only for learning.	Logical backup/ Snapshot backup **NOTE** • This function is now in OBT. To use it, submit a service ticket. • For details about how to set **Backup Method** to **Snapshot**, see **Setting Backup Method for a DB Instance**.	Full backup

Pricing

- After you purchase an instance, DDS will provide additional backup storage of the same size as you purchased. For example, if you purchase 100 GB of instance storage space, you will obtain 100 GB of backup storage space. If the size of backup data does not exceed 100 GB, the backup data is stored on OBS free of charge. If the size of the backup data exceeds 100 GB, you will be charged based on the **OBS billing rules**.
- You can check your expenditure records for DDS backup fees by going to **Billing Center** > **Bills**.

Precautions

- The backup process does not affect services.
- DDS checks existing automated backup files. If the retention period of a file exceeds the backup retention period you set, DDS will delete the file.
- After the backup policy is modified, an automated backup will be triggered based on the new backup policy. The retention period of the previously generated automated backups remains unchanged.
- Single node instances do not support incremental backup.

Enabling or Modifying an Automated Backup Policy

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the navigation pane on the left, choose **Backups & Restorations**.

Step 6 On the **Backups & Restorations** page, click **Set Backup Policy**. If you want to enable the automated backup policy, click . Once enabled, the backup policy can be modified as shown in **Figure 8-4**.

Figure 8-4 Set Backup Policy

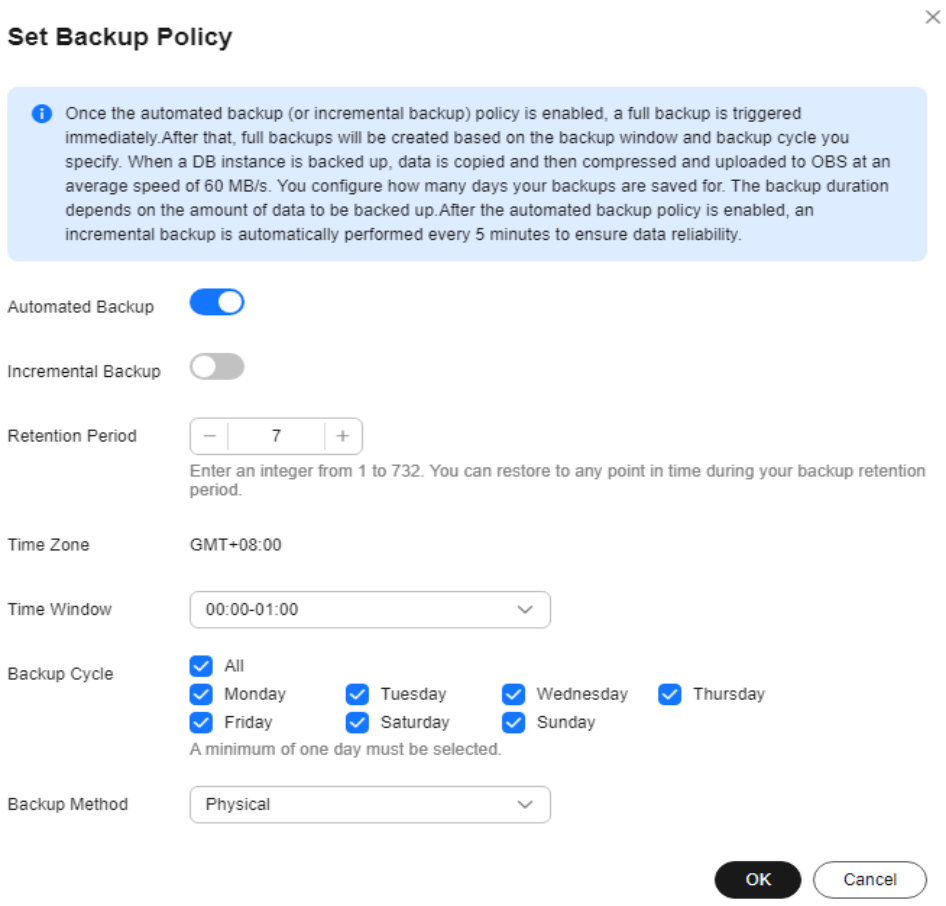


Table 8-5 Parameter description

Parameter	Description
Retention Period (days)	Number of days that your automated backups can be retained. The retention period is from 1 to 732 days and the default value is **7**. ● Extending the retention period improves data reliability. You can configure the retention period if needed. ● If you shorten the retention period, the new backup policy takes effect for existing backups. Any automated backups (including full and incremental backups) that have expired will be automatically deleted. Manual backups will not be automatically deleted but you can delete them manually.
Time Zone	The default backup time zone is the UTC time.
Time Window	A one-hour period the backup will be scheduled within 24 hours, such as 01:00-02:00. The backup time is in UTC format.
Backup Cycle	● If you set the retention period to 1 to 6 days, data is automatically backed up each day of the week and the backup cycle cannot be changed. ● If you set the retention period to 7 to 732 days, you must select at least one day of the week for the backup cycle.
Backup Method	● **Physical**: Data is copied from physical disks. ● **Snapshot**: The data status at a particular point in time is retained. Compared with physical backup, snapshot backup is faster. ● **Logical**: A tool is used to read data and logically export the data.

Step 7 Click **OK** to save the changes.

Step 8 View the results.

- During the creation of an automated backup, you can query the backup status on the **Backups** page or the **Backups & Restorations** tab. The backup status is **Backing up**.
- In the upper right corner of the backup list, click to refresh the list. The backup status changes to **Complete**. The backup type is **Automated** and the backup method is **Physical**.

----End

Disabling an Automated Backup Policy

NOTICE

When disabling the automated backup policy:

- Your data cannot be backed up.
- Your replica set instances cannot be restored to a specified point in time.
- If you choose to delete all the existing automated backup when disabling the automated backup policy, related restoration or download operations will fail.

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

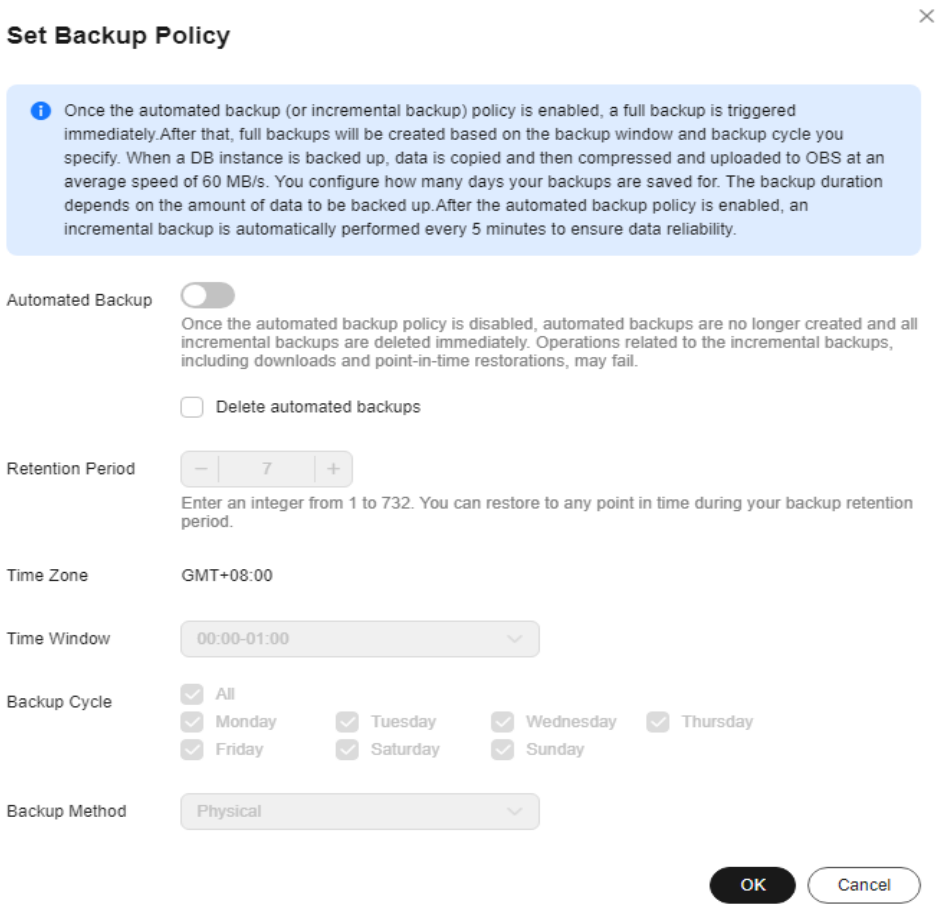
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the navigation pane on the left, choose **Backups & Restorations**.

Step 6 On the **Backups & Restorations** page, click **Set Backup Policy**. On the displayed page, click to disable the automated backup policy. **Figure 8-5** shows the dialog box for modifying the backup policy.

Figure 8-5 Set Backup Policy



You can determine whether to delete all automated backup files:

- If you do not select **Delete automated backups**, all backup files within the retention period will be retained, but you can still delete them manually. For details, see section **Deleting an Automated Backup**.
- If you select **Delete automated backups**, all backup files within the retention period will be deleted.

If you shorten the retention period, the new backup policy takes effect for all backup files. Any backup files that have expired, based on a newly configured retention period, will be deleted, but the latest expired backup file will be retained.

Step 7 Click **OK**.

NOTE

- If automated backups are disabled, any automated backups in progress stop immediately.
- After automated backups are disabled, incremental backups are disabled by default.
- If you need to enable the automated backup policy again, see **Enabling or Modifying an Automated Backup Policy**.

----End

8.3.2 Configuring an Incremental Backup Policy

Scenarios

Incremental backup is used to back up the data newly added or modified since the last full or incremental backup. DDS automatically backs up the updated data every 5-60 minutes since the last automated or incremental backup was made.

When you create a DDS DB instance, incremental backup is enabled by default for all DB instances except DB instances with fewer than 4 vCPUs. You can enable or disable an incremental backup policy after an instance is created. For details, see **Enabling or Modifying an Incremental Backup Policy** and **Disabling the Incremental Backup Policy**.

After an incremental backup policy is enabled for a DDS instance, incremental files are not displayed on the DDS console.

Prerequisites

Before enabling the incremental backup policy, ensure that the automated backup policy has been enabled. For details, see **Enabling or Modifying an Automated Backup Policy**.

Precautions

- Incremental backup is performed on the hidden node. After an incremental backup policy is enabled, the CPU and memory usage of the hidden node increases, depending on the service model.
- To ensure stable database running, you are advised to double the specifications of the hidden node when the CPU usage exceeds 60% or the memory usage exceeds 80%.

Constraints

- Only cluster instances of version 4.0 or later and replica set instances of version 3.4 or later support this function.
- To minimize the impact of incremental backup on cluster instances, incremental backup is disabled by default for cluster instances with mongos nodes with fewer than 4 vCPUs.
- Incremental backup stops in any of the following scenarios and starts again after the next automated backup is complete:
  - rename operation
  - collmod operation
  - Creating a user
  - Deleting a user
  - Creating a role
  - Deleting a role
  - Enabling shard IP addresses of a cluster instance
  - Changing the password of the shard node user
  - Enabling config IP addresses of a cluster instance

- Changing the password of the config node user
- Changing the password of the **rwuser** user

Enabling or Modifying an Incremental Backup Policy

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the navigation pane on the left, choose **Backups & Restorations**.

Step 6 On the **Backups & Restorations** page, click **Set Backup Policy**. To enable incremental backup, click  . After incremental backup is enabled, a full backup is triggered.

Table 8-6 Parameter description

Parameter	Description
Automated Backup	For details about automated backup parameters, see **Table 8-5**.
Incremental Backup	Before enabling the incremental backup policy, ensure that the automated backup policy has been enabled.

Step 7 Click **OK**.

Step 8 View the results.

- During the creation of an automated backup, you can query the backup status on the **Backups** page or the **Backups & Restorations** tab. The backup status is **Backing up**.
- In the upper right corner of the backup list, click  to refresh the list. The backup status changes to **Complete**.

----End

Disabling the Incremental Backup Policy

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the navigation pane on the left, choose **Backups & Restorations**.

Step 6 On the **Backups & Restorations** page, click **Set Backup Policy**.

Step 7 In the displayed dialog box, click  to the right of **Incremental Backup** to disable the incremental backup policy.

Step 8 Click **OK**.

NOTICE

- After you disable this incremental backup, the incremental backup task will be stopped, all incremental backup files will be deleted immediately, and operations related to incremental backup fail.
- After a DB instance is deleted, all incremental backup files of the DB instance are retained. The retention period depends on the incremental backup retention period that you specified.

----End

8.3.3 Setting Backup Method for a DB Instance

DDS allows snapshot backup for a DB instance.

NOTE

Huawei Cloud has discontinued the sale of DDS single node instances since July 15, 2023.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the navigation pane on the left, choose **Backups & Restorations**.

Step 6 On the **Backups & Restorations** page, click **Set Backup Policy**.

Figure 8-6 Setting backup method for a DB instance

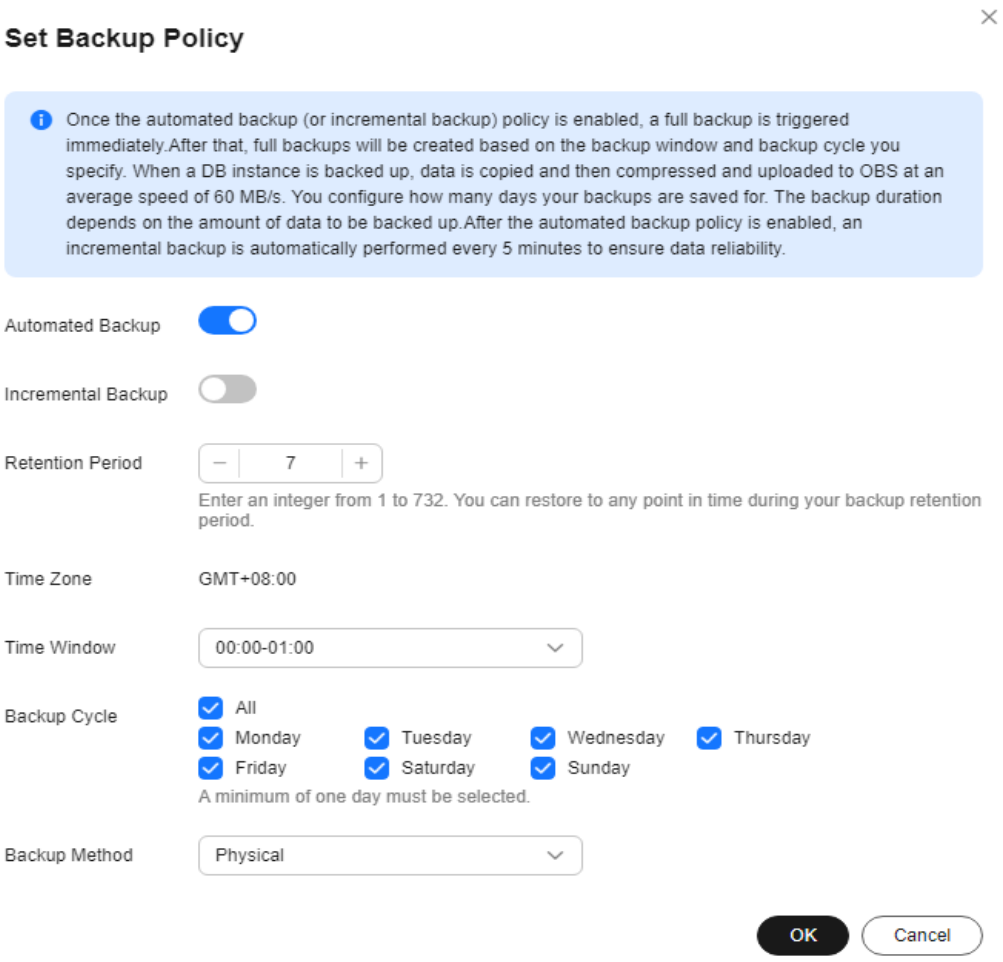


Table 8-7 Parameter description

Parameter	Description
Retention Period (days)	Number of days that your automated backups can be retained. The retention period is from 1 to 732 days and the default value is **7**. • Extending the retention period improves data reliability. You can configure the retention period if needed. • If you shorten the retention period, the new backup policy takes effect for existing backups. Any automated backups (including full and incremental backups) that have expired will be automatically deleted. Manual backups will not be automatically deleted but you can delete them manually.
Time Zone	The default backup time zone is the UTC time.
Time Window	The backup interval is one hour. You are advised to set the backup window to an off-peak period.

Parameter	Description
Backup Cycle	• If you set the retention period to 1 to 6 days, data is automatically backed up each day of the week and the backup cycle cannot be changed. • If you set the retention period to 7 to 732 days, you must select at least one day of the week for the backup cycle.
Backup Method	• **Physical**: Data is copied from physical disks. • **Snapshot**: The data status at a particular point in time is retained. Compared with physical backup, snapshot backup is faster. **NOTE** This function is now in OBT. To use it, submit a service ticket. • **Logical**: A tool is used to read data and logically export the data.

Step 7 Set **Backup Method** to **Snapshot** and click **OK**.

----End

8.3.4 Creating a Manual Backup

Scenarios

This section describes how to create a manual backup. Creating a backup for a DB instance helps ensure data can be restored if needed, ensuring data reliability.

Prerequisites

You can create backups (including manual backups, automated backups, and incremental backups) only when the hidden nodes of cluster instances and replica set instances are normal.

Description

- Backup type

  Full backup: All data is backed up even if no data is updated since the last backup.
- Backup mode

  Physical backup: Data is copied from physical disks.
- Table 8-8 lists the manual backup methods supported by DDS.

Table 8-8 Backup methods

Instance Type	Backup Mode	Backup Type
Cluster	Physical backup	Full backup

Instance Type	Backup Mode	Backup Type
Replica set	Physical backup	Full backup
Single node **NOTE** Single node instances apply to only a few scenarios. You are advised to use a single node instance only for learning.	Logical backup	Full backup

Pricing Details

- After you purchase an instance, DDS will provide additional backup storage of the same size as you purchased. For example, if you purchase 100 GB of instance storage space, you will obtain 100 GB of backup storage space. If the size of backup data does not exceed 100 GB, the backup data is stored on OBS free of charge. If the size of the backup data exceeds 100 GB, you will be charged based on the **OBS billing rules**.
- You can check your expenditure records for DDS backup fees by going to **Billing Center** > **Bills**.
- Backups that are not normally delivered by a customer (for example, full backups automatically delivered after node rebuilding) are not displayed on the **Backups** page because they are not billed.

Precautions

- The backup process does not affect services.
- When you delete a DB instance, its automated backups are also deleted but its manual backups are retained.

Procedure

Step 1 Log in to the management console.

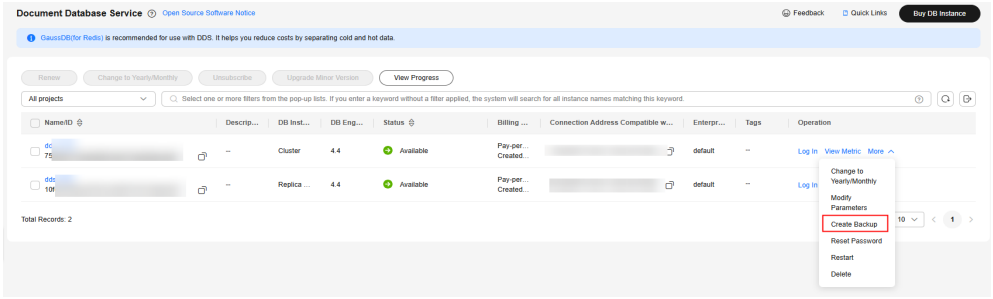
Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 Create a manual backup on the DDS console in any of the following ways:

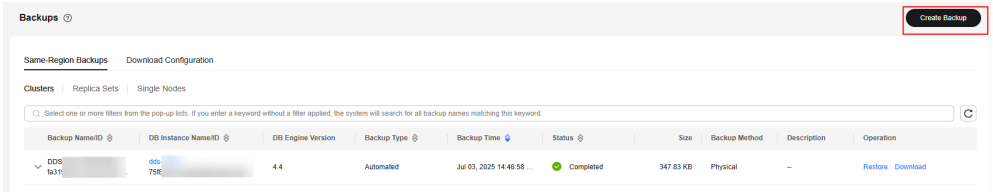
- On the **Instances** page, locate an available instance and choose **More** > **Create Backup** in the **Operation** column.

Figure 8-7 Method 1: Creating a backup



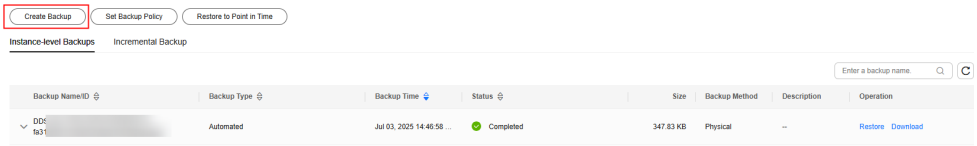
- On the **Instances** page, choose **Backups** in the navigation pane on the left. On the displayed page, click **Create Backup**.

Figure 8-8 Method 2: Creating a backup



- On the **Instances** page, click an available DB instance. In the navigation pane on the left, choose **Backups & Restorations**. On the **Backups & Restorations** page, click **Create Backup**.

Figure 8-9 Method 3: Creating a backup



Step 5 In the displayed dialog box, specify **Backup Name** and **Description** and click **OK**.

- The manual backup name can be 4 to 64 characters long. It must start with a letter and can contain only letters, digits, hyphens (-), and underscores (_).
- The description contains a maximum of 256 characters and cannot contain the carriage return character and the following special characters: >!<"&'=

Step 6 View the results.

- During the creation of a manual backup, you can query the backup status on the **Backups** or the **Backups & Restorations** page. The backup status is **Backing up**. The time it takes to complete the backup depends on the size of the job.
- If the manual backup is successfully created, the backup status is **Complete**. The manual backup type is **Manual** and the backup method is **Physical**.

----End

8.4 Managing Backups

8.4.1 Deleting a Manual Backup

Scenarios

This section describes how to delete manual backups to release the storage space.

Precautions

- Deleted backups cannot be restored. Exercise caution when performing this operation.
- Backups being used to recover instances cannot be deleted.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

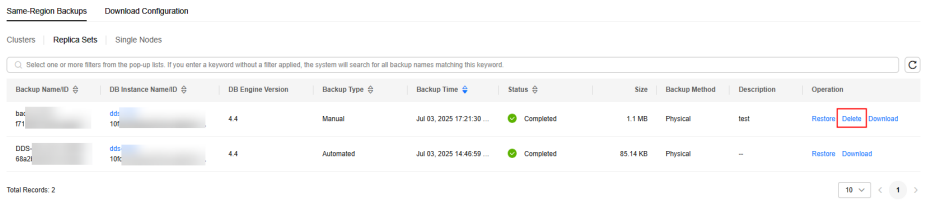
Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 Delete a manual backup.

On the DDS console, you can delete a manual backup using either of the following methods:

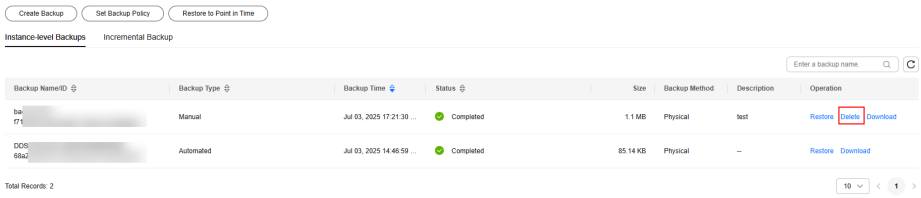
- Method 1:
  a. In the navigation pane on the left, choose **Backups**.
  b. On the **Backups** page, click the **Clusters**, **Replica Sets**, or **Single Nodes** tab.
  c. Locate the manual backup to be deleted and click **Delete** in the **Operation** column.

  Figure 8-10 Deleting a manual backup

- Method 2:
  a. On the **Instances** page, click the target DB instance.
  b. In the navigation pane on the left, choose **Backups & Restorations**.
  c. On the **Backups & Restorations** page, locate the manual backup to be deleted and click **Delete**.

Figure 8-11 Deleting a Manual Backup



Step 5 In the displayed dialog box, click **Yes**.

----End

8.4.2 Deleting an Automated Backup

Scenarios

DDS allows you to delete failed automated backups to release storage space. Deleted backups cannot be restored. Exercise caution when performing this operation.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

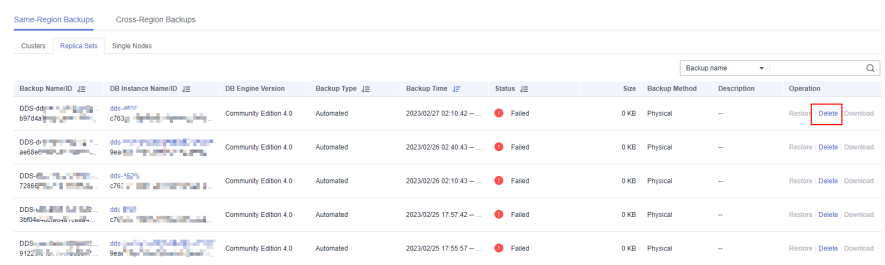
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 Delete an automated backup.

On the DDS console, you can delete an automated backup using either of the following methods:

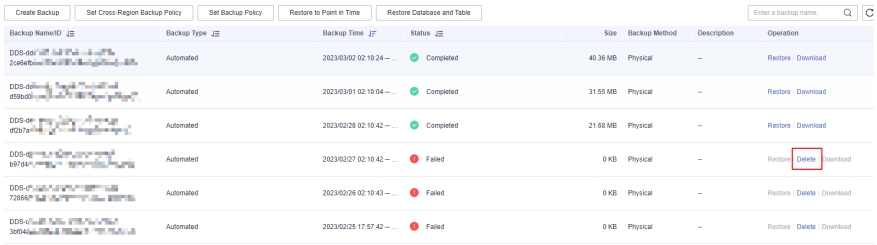
- Method 1:
  a. In the navigation pane on the left, choose **Backups**.
  b. On the **Backups** page, click the **Clusters**, **Replica Sets**, or **Single Nodes** tab.
  c. Locate the automated backup to be deleted and click **Delete** in the **Operation** column.

Figure 8-12 Deleting an automated backup



- Method 2:
  a. On the **Instances** page, click the instance name.
  b. In the navigation pane on the left, choose **Backups & Restorations**.
  c. On the **Backups & Restorations** tab, locate the automated backup to be deleted and click **Delete** in the **Operation** column.

  Figure 8-13 Deleting an automated backup



Step 5 In the displayed dialog box, click **Yes**.

----End

8.4.3 Downloading a Backup File

8.4.3.1 Using OBS Browser+

Scenarios

You can use OBS Brower+ to download a manual or an automated backup to a local device for backup or restoration.

Precautions

- When you use OBS Browser+ to download backup data, you will not be billed for outbound traffic from OBS.
- If the size of a backup file is greater than 400 MB, use OBS Browser+ to download the backup file.
- Backups downloaded from the DDS console are all full backups.

Procedure

Step 1 Log in to the management console.

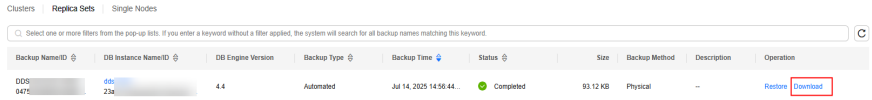
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Backups**.

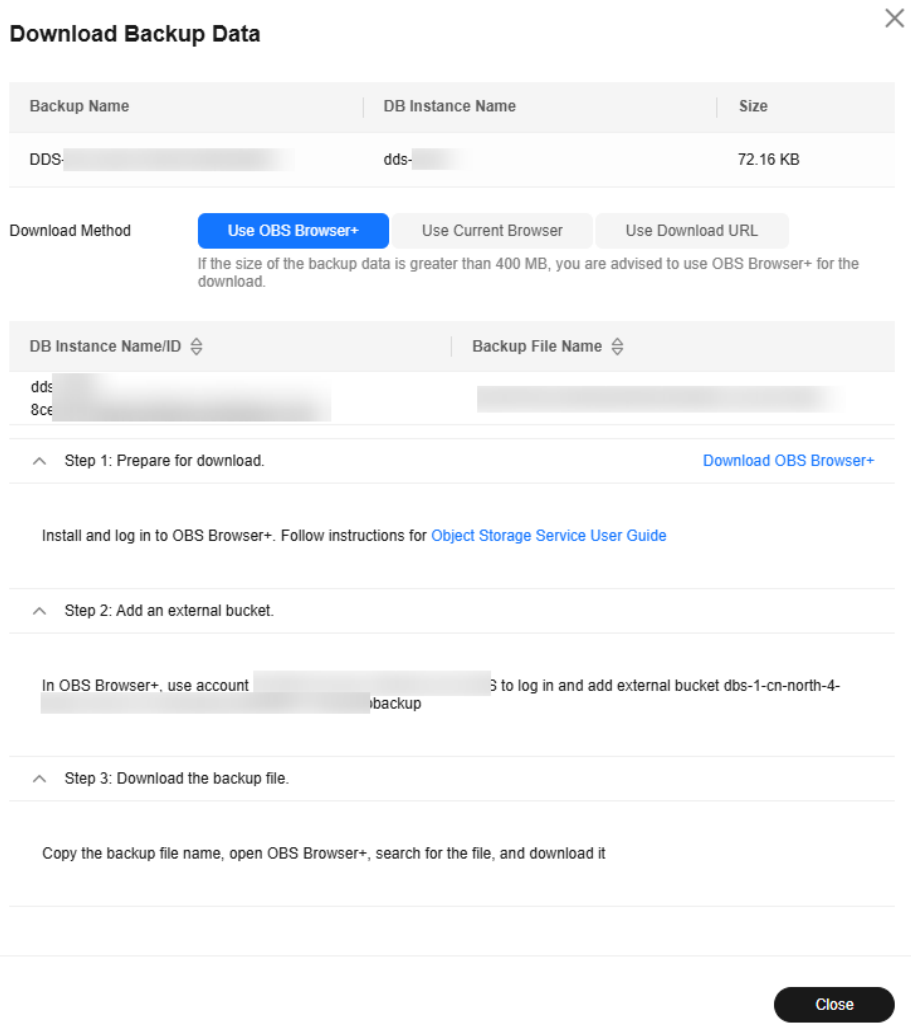
Step 5 On the **Backups** page, click the **Clusters**, **Replica Sets**, or **Single Nodes** tab, locate the available backup you want to download and click **Download** in the **Operation** column.

Figure 8-14 Download Backup



Step 6 In the displayed dialog box, select **Use OBS Browser+** for **Download Method** and download the backup as prompted.

Figure 8-15 Selecting a download method



Step 7 Download OBS Browser+ by clicking **Download OBS Browser+** in Step 1 on **Figure 8-15**.

Step 8 Decompress and install OBS Browser+.

Step 9 Log in to OBS Browser+.

For details about how to log in to OBS Browser+, see "Logging In to OBS Browser +" in the *Object Storage Service Tools Guide*.

Step 10 Add an external bucket.

In the **Add External Bucket** dialog box of OBS Browser+, enter the bucket name displayed in Step 2 on **Figure 8-15**, and click **OK**.

Step 11 Download the backup file.

On the OBS Browser+ page, click the external bucket that you added. In the search box on the right of OBS Browser+, enter the backup file name displayed in Step 3 on **Figure 8-15**. In the search result, locate the target backup and download it.

Step 12 After the backup file is downloaded, use the LZ4 to decompress the file.

Run the following command to decompress the backup file:

lz4 -d *$1* **| tar -xC** *$2*

$1: indicates the downloaded backup file.

$2: indicates the directory to which the backup file is decompressed.

Step 13 You can restore data locally as required.

For details, see the following documentation.

- **Restoring a Cluster Backup to an On-premises Database**
- **Restoring a Replica Set Backup to an On-Premises Database**

----End

8.4.3.2 Using Current Browser

Scenarios

You can use a browser to download a manual or an automated backup to a local device for backup or restoration.

Precautions

- Cluster backup files cannot be downloaded using a browser.
- Backups downloaded from the DDS console are all full backups.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

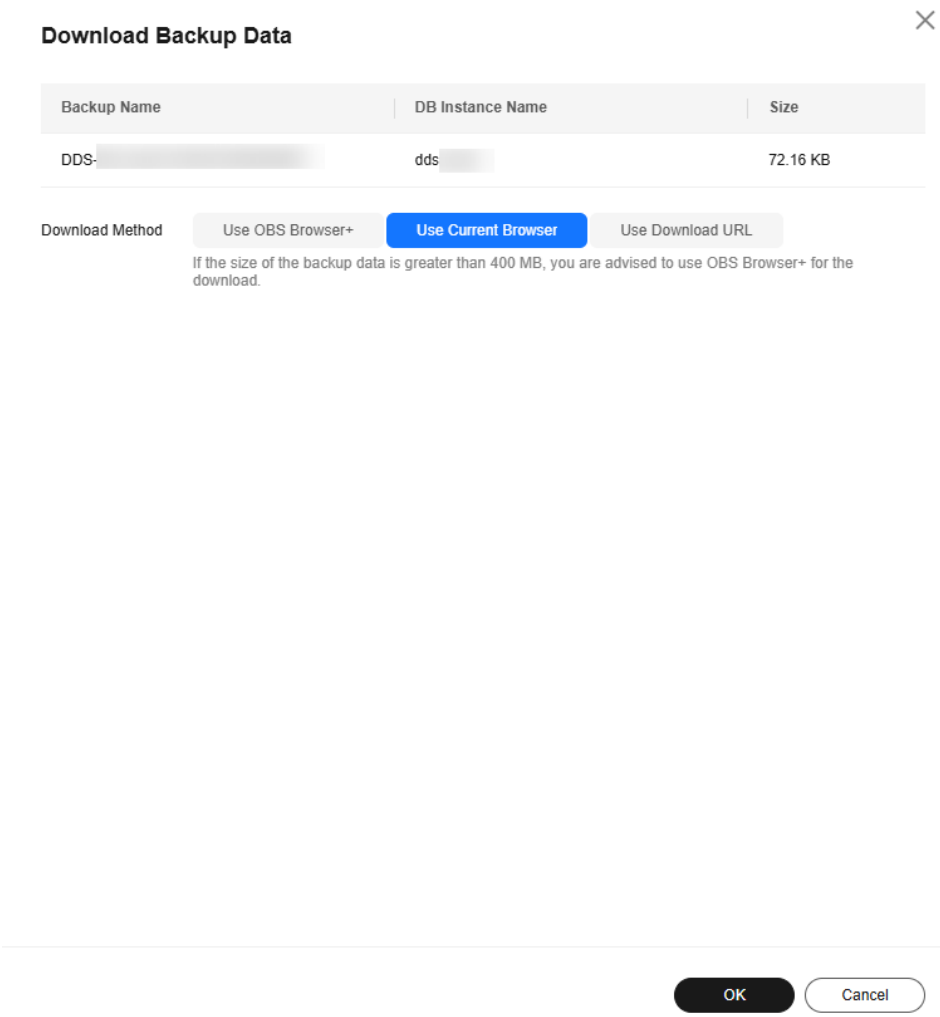
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Backups**.

Step 5 On the **Backups** page, click the **Clusters**, **Replica Sets**, or **Single Nodes** tab, locate the available backup you want to download and click **Download** in the **Operation** column.

Step 6 In the displayed dialog box, select **Use Current Browser** for **Download Method** and click **OK**.

Figure 8-16 Selecting a download method



Step 7 After the backup file is downloaded, decompress it using LZ4.

Run the following command to decompress the backup file:

lz4 -d *$1* **| tar -xC** *$2*

$1: indicates the downloaded backup file.

$2: indicates the directory to which the backup file is decompressed.

Step 8 You can restore data locally as required.

For details, see the following documentation.

- **Restoring a Cluster Backup to an On-premises Database**
- **Restoring a Replica Set Backup to an On-Premises Database**

----End

8.4.3.3 Using Download URL

Scenarios

You can download manual or automated backup files using the URL provided by DDS to a local device for backup or restoration.

Precautions

Backups downloaded from the DDS console are all full backups.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

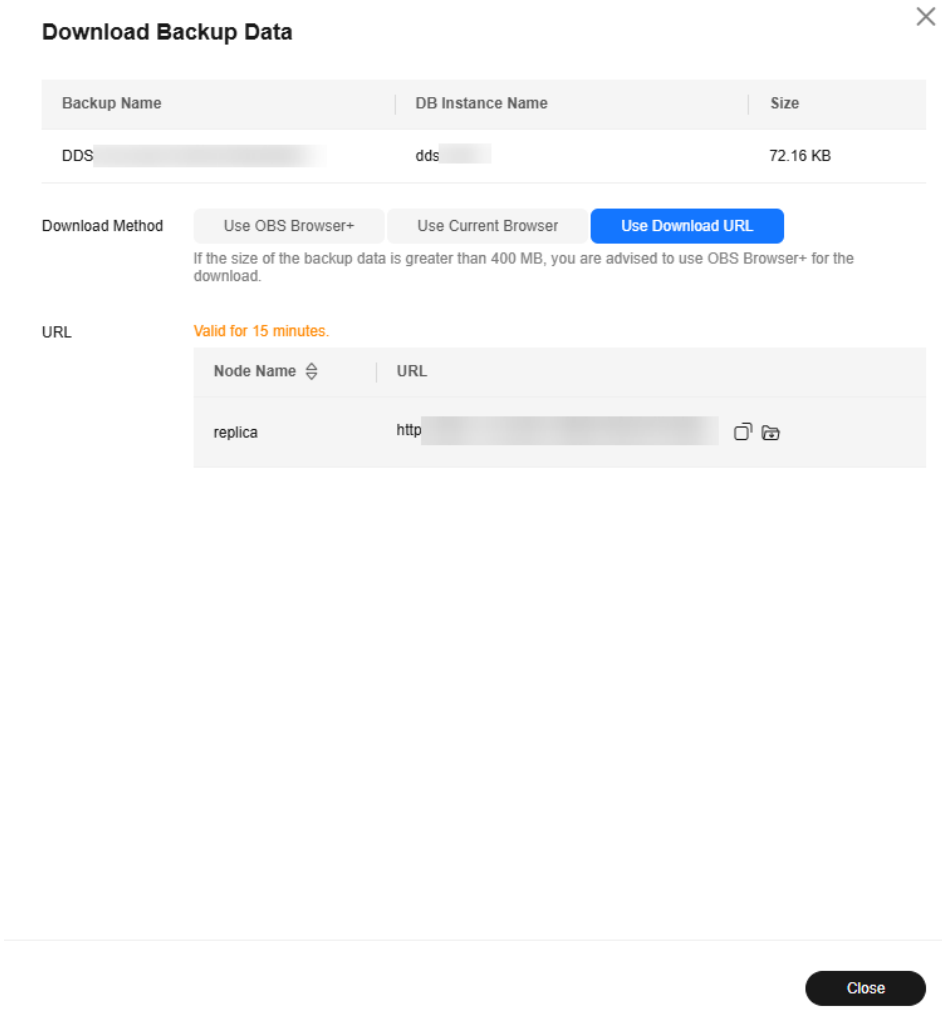
Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Backups**.

Step 5 On the **Backups** page, click the **Clusters**, **Replica Sets**, or **Single Nodes** tab, locate the available backup you want to download and click **Download** in the **Operation** column.

Step 6 In the displayed dialog box, select **Use Download URL** for **Download Method**, click  to copy the URL, and click **OK**.

Figure 8-17 Selecting a download method



A valid URL for downloading the backup data is displayed.

- You can use various download tools, such as your browser and Xunlei to download backup files.
- You can also run the following command to download backup files:

  wget -O *FILE_NAME* **--no-check-certificate "***DOWNLOAD_URL***"**

  Parameter description:

  FILE_NAME is the new name of the downloaded backup file. The original backup file name may be too long and exceed the maximum characters allowed by the client file system, so you are advised to rename the backup file.

  DOWNLOAD_URL is the location of the backup file to be downloaded. If the location contains special characters, escape is required.

Step 7 After the backup file is downloaded, decompress it using LZ4.

Run the following command to decompress the backup file:

lz4 -d *$1* **| tar -xC** *$2*

$1: indicates the downloaded backup file.

$2: indicates the directory to which the backup file is decompressed.

Step 8 You can restore data locally as required.

For details, see the following documentation.

- **Restoring a Cluster Backup to an On-premises Database**
- **Restoring a Replica Set Backup to an On-Premises Database**

----End

9 Data Restorations

9.1 Solutions

DDS provides multiple data restoration solutions. You can select a proper solution to meet your service requirements.

NOTE

By default, all DDS versions 3.2, 3.4, 4.0, 4.2, and 4.4 are supported unless otherwise specified.

Table 9-1 Solutions

Restoration Type	Instance Type and Version	Scenario
Restoring Data to a New Instance	• Cluster • Replica set • Single node	You can restore an existing automated or manual backup file to a new instance.
Restoring Data to the Original Instance	• Cluster • Replica set • Single node	You can restore an existing automated or manual backup file to the original instance.
Restoring Data to a Point in Time	• Cluster 4.0 or later • Replica set 4.0 or later	You can restore an instance to a point in time.
Restoring Database Tables to a Point in Time	• Replica set 4.0 or later • Cluster 4.0 or later	You can restore a database table to a point in time.

Restoration Type	Instance Type and Version	Scenario
Restoring Data to an On-Premises Database	• Cluster (versions 3.4 and 4.0) • Replica set (versions 3.4 and 4.0) • Single node (versions 3.4 and 4.0)	You can download a DDS backup file to your local PC and restore data to an on-premises database.
Restoring Data Using mongorestore	• Cluster • Replica set • Single node	You can use tools provided by the MongoDB client to restore data.
Restoring Data Using mongoimport	• Cluster • Replica set • Single node	You can use tools provided by the MongoDB client to restore data.

9.2 Restoring Data to a New Instance

9.2.1 Restoring a Cluster Backup to a New Instance

Scenarios

DDS allows you to restore an existing automated or manual backup to a new instance. The restored data is the same as the backup data.

When you restore an instance from a backup file, a full backup file is downloaded from OBS and then restored to the instance at an average speed of 40 MB/s.

Precautions

To restore backup files to a new instance, your account balance must be greater than or equal to $0 USD. You will pay for the new instance specifications.

Procedure

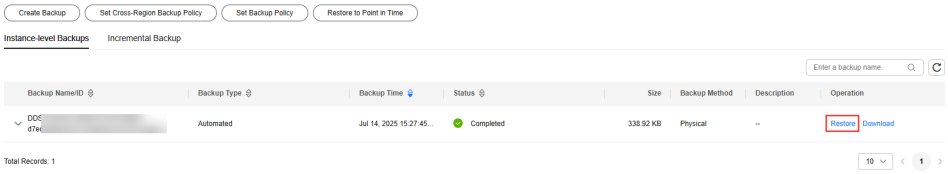
Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

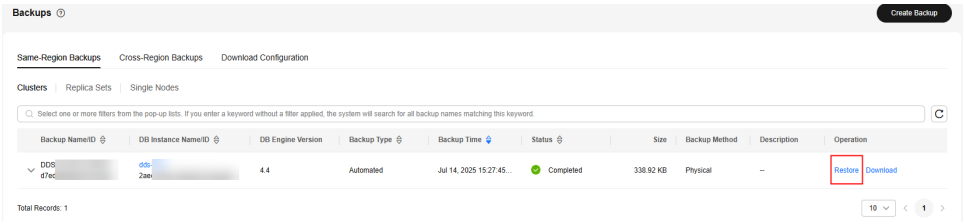
Step 4 On the **Instances** page, click the cluster instance name. Choose **Backups & Restorations** in the navigation pane, select the backup to be restored, and click **Restore**.

Figure 9-1 Restoring a cluster from a backup



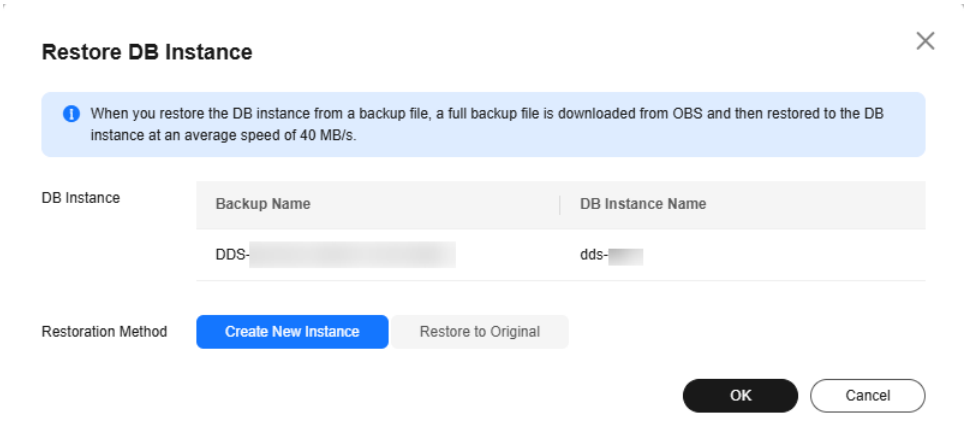
Alternatively, on the navigation pane on the left, choose **Backups**. On the **Backups** page, locate the target backup on the **Clusters** tab and click **Restore** in the **Operation** column.

Figure 9-2 Restoring a cluster from a backup



Step 5 In the **Restore DB Instance** dialog box, select **Create New Instance** for **Restoration Method** and click **OK**.

Figure 9-3 Restoring a cluster backup to a new instance



Step 6 The **Create New Instance** page is displayed for you to create an instance using the backup data. The new DB instance is independent of the original one.

- You are recommended to deploy the restored DB instance in a different AZ to ensure that applications will not be adversely affected by the failure in any single AZ.
- The database type, DB instance type, compatible MongoDB version, storage engine, storage type, and shard quantity must be the same as those of the original and cannot be changed.

- The number of dds mongos nodes is 2 by default and ranges from 2 to 16. You can specify the quantity.
- The storage space is the same as that of the original shard node by default. You can increase the storage space, but you cannot reduce it.
- Other settings are the same as those of the original DB instance by default and can be modified. For details, see **Buying a Cluster Instance**.
- A full backup is triggered after the new instance is created.

----End

9.2.2 Restoring a Replica Set Backup to a New Instance

Scenarios

DDS allows you to restore an existing automated or manual backup to a new instance. The restored data is the same as the backup data.

When you restore an instance from a backup file, a full backup file is downloaded from OBS and then restored to the instance at an average speed of 40 MB/s.

Precautions

To restore backup files to a new instance, your account balance must be greater than or equal to $0 USD. You will pay for the new instance specifications.

Procedure

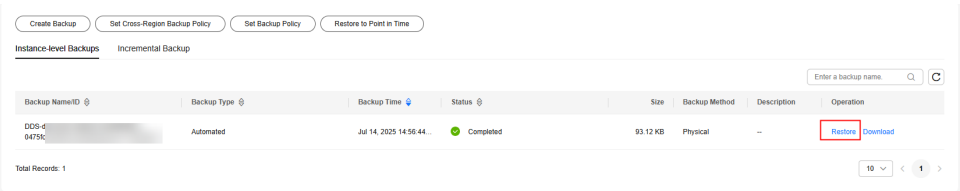
Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

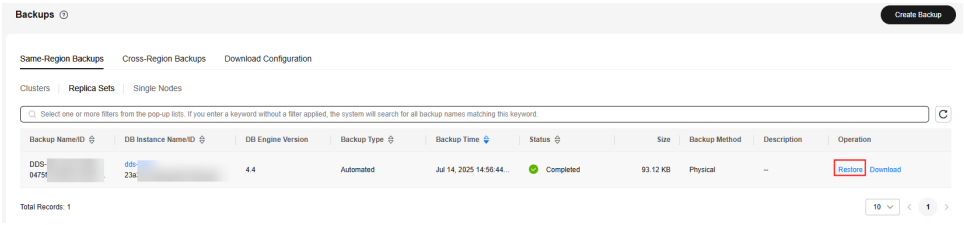
Step 4 On the **Instances** page, click the replica set instance. Choose **Backups & Restorations** in the navigation pane, select the backup to be restored, and click **Restore**.

Figure 9-4 Restoring a replica set instance backup



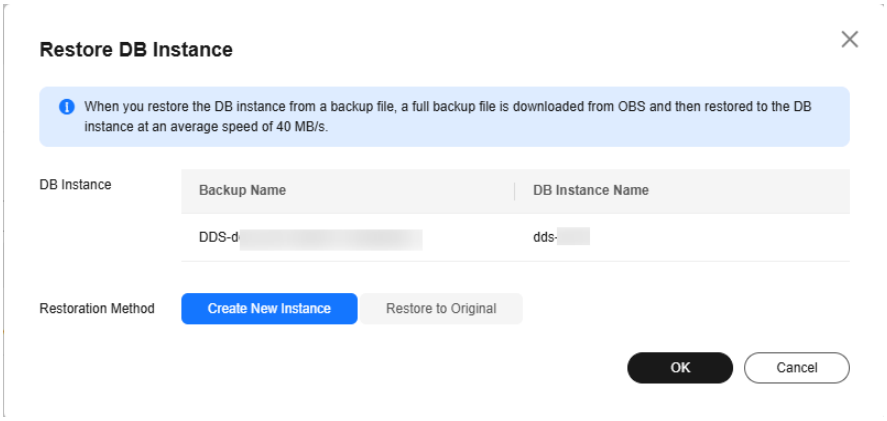
Alternatively, on the navigation pane on the left, choose **Backups**. On the **Backups** page, locate the backup on the **Replica Sets** tab and click **Restore** in the **Operation** column.

Figure 9-5 Restoring a replica set instance backup



Step 5 In the **Restore DB Instance** dialog box, select **Create New Instance** for **Restoration Method** and click **OK**.

Figure 9-6 Restoring to a new instance



Step 6 The **Create New Instance** page is displayed for you to create an instance using the backup data. The new DB instance is independent of the original one.

- You are recommended to deploy the restored DB instance in a different AZ to ensure that applications will not be adversely affected by the failure in any single AZ.
- The database type, DB instance type, compatible MongoDB version, storage engine, and storage type must be the same as those of the original and cannot be changed.
- The storage space is the same as that of the original instance by default. You can increase the storage space, but you cannot reduce it.
- Other settings have default values and can be modified. For details, see **Buying a Replica Set Instance**.
- A full backup is triggered after the new instance is created.

----End

9.2.3 Restoring a Single Node Backup to a New Instance

Scenarios

DDS allows you to restore an existing automated or manual backup to a new instance. The restored data is the same as the backup data.

When you restore an instance from a backup file, a full backup file is downloaded from OBS and then restored to the instance at an average speed of 40 MB/s.

> **NOTE**
>
> Huawei Cloud has discontinued the sale of DDS single node instances since July 15, 2023.

Precautions

To restore backup files to a new instance, your account balance must be greater than or equal to $0 USD. You will pay for the new instance specifications.

Procedure

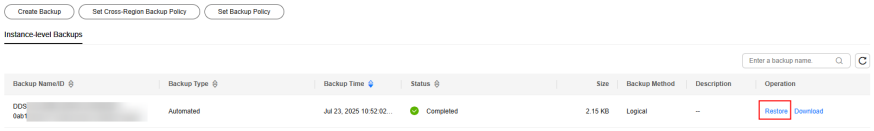
Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

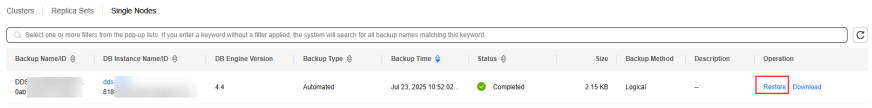
Step 4 On the **Instances** page, click the single node instance name. Choose **Backups & Restorations** in the navigation pane, select the backup to be restored, and click **Restore**.

Figure 9-7 Restoring a single node backup



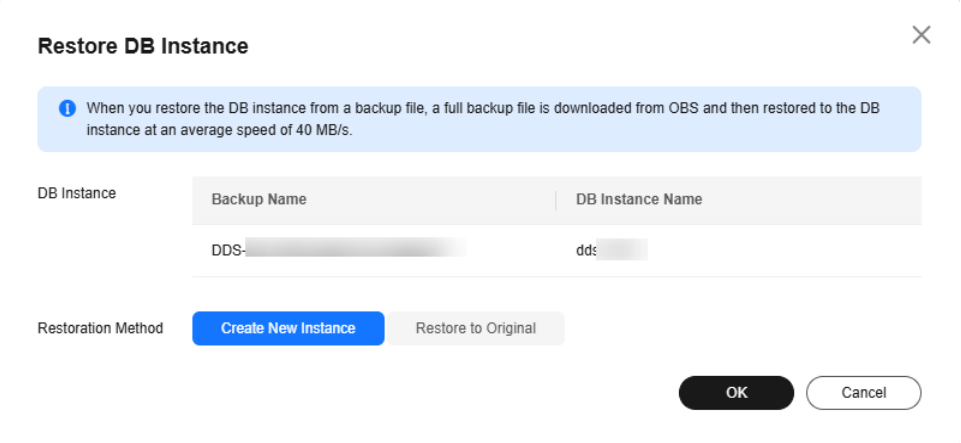
Alternatively, on the navigation pane on the left, choose **Backups**. On the **Backups** page, locate the target backup on the **Single Nodes** tab and click **Restore** in the **Operation** column.

Figure 9-8 Restoring a single node backup



Step 5 In the **Restore DB Instance** dialog box, select **Create New Instance** for **Restoration Method** and click **OK**.

Figure 9-9 Restoring a single node backup to a new instance



Step 6 The **Create New Instance** page is displayed for you to create an instance using the backup data. The new DB instance is independent of the original one.

- You are recommended to deploy the restored DB instance in a different AZ to ensure that applications will not be adversely affected by the failure in any single AZ.
- The database type, DB instance type, compatible MongoDB version, storage engine, and storage type must be the same as those of the original and cannot be changed.
- The storage space is the same as that of the original instance by default. You can increase the storage space, but you cannot reduce it.
- A full backup is triggered after the new instance is created.

----End

9.3 Restoring Data to the Original Instance

9.3.1 Restoring a Cluster Backup to the Original Instance

Scenarios

DDS allows you to restore an existing automated or manual backup to an original instance. The restored data is the same as the backup data.

When you restore an instance from a backup file, a full backup file is downloaded from OBS and then restored to the instance at an average speed of 40 MB/s.

Precautions

- Restoring backup data to the original instance will overwrite existing data on the instance and cause the instance to be unavailable during the restoration. Exercise caution when performing this operation.
- The administrator password of the instance remains unchanged after the restoration.

- If you restore a manual backup, check whether the instance to which the manual backup belongs exists. If the instance does not exist, the backup can only be restored to a new instance.
- If a cluster DB instance have read replicas associated, backup data can only be restored to a new DB instance.

Procedure

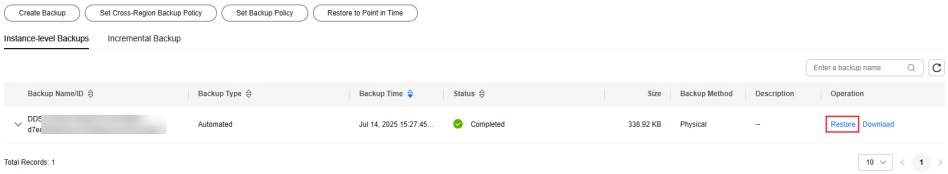
Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

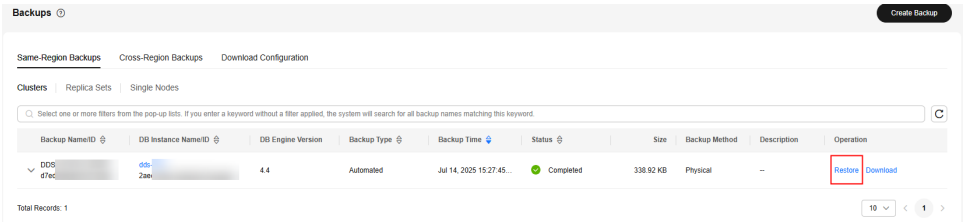
Step 4 On the **Instances** page, click the cluster instance name. Choose **Backups & Restorations** in the navigation pane on the left, select the backup to be restored, and click **Restore**.

Figure 9-10 Restoring a cluster from a backup



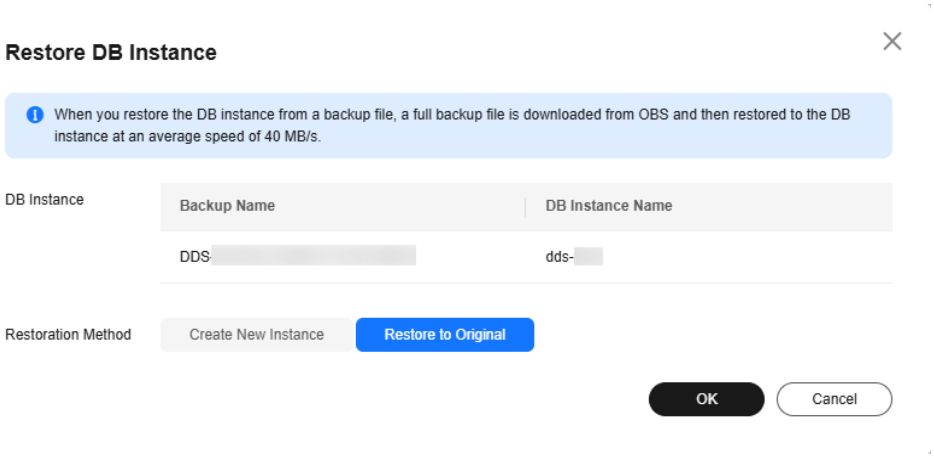
Alternatively, on the navigation pane on the left, choose **Backups**. On the **Backups** page, locate the target backup on the **Clusters** tab and click **Restore** in the **Operation** column.

Figure 9-11 Restoring a cluster from a backup



Step 5 In the **Restore DB Instance** dialog box, select **Restore to Original** for **Restoration Method** and click **OK**.

Figure 9-12 Restore to Original



- On the **Instances** page, the status of the instance changes from **Restoring** to **Available**.
- After the restoration is complete, a full backup will be automatically triggered.

----End

9.3.2 Restoring a Replica Set Backup to the Original Instance

Scenarios

DDS allows you to restore an existing automated or manual backup to an original instance. The restored data is the same as the backup data.

When you restore an instance from a backup file, a full backup file is downloaded from OBS and then restored to the instance at an average speed of 40 MB/s.

Precautions

- Restoring backup data to the original instance will overwrite existing data on the instance and cause the instance to be unavailable during the restoration. Exercise caution when performing this operation.
- The administrator password of the instance remains unchanged after the restoration.
- If you restore a manual backup, check whether the instance to which the manual backup belongs exists. If the instance does not exist, the backup can only be restored to a new instance.

Procedure

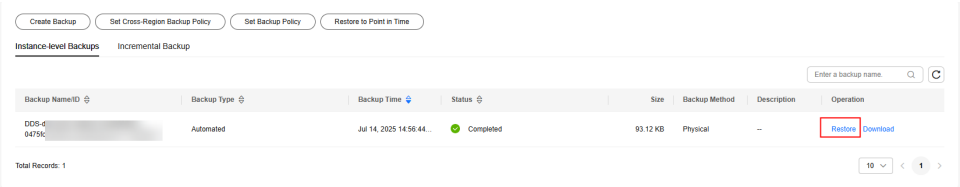
Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

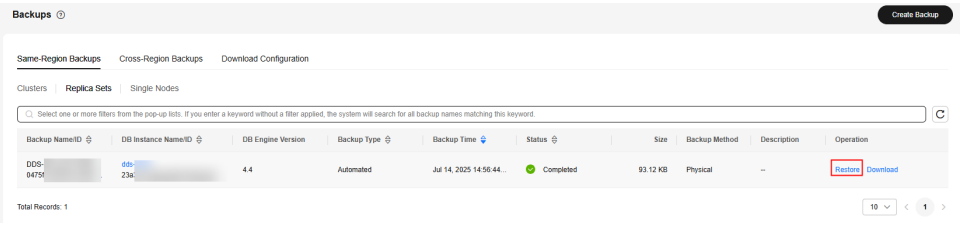
Step 4 On the **Instances** page, click the replica set instance. Choose **Backups & Restorations** in the navigation pane on the left, select the backup to be restored, and click **Restore**.

Figure 9-13 Restoring a replica set instance backup



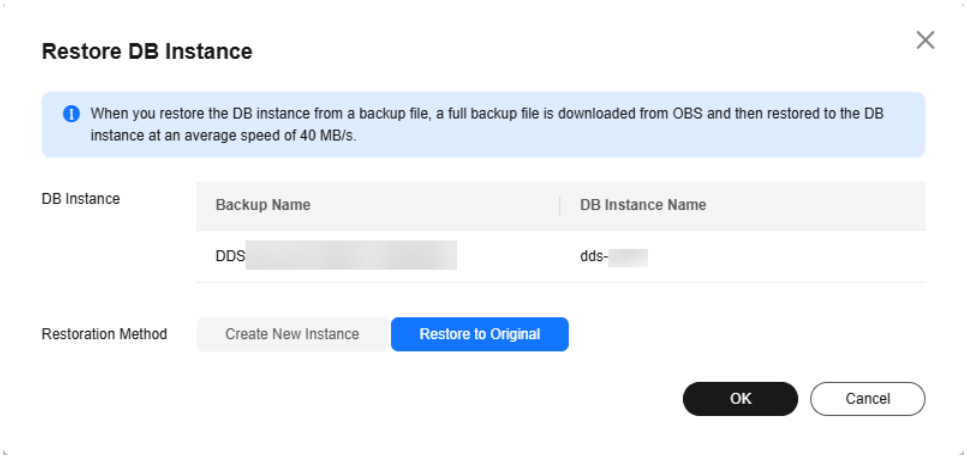
Alternatively, on the navigation pane on the left, choose **Backups**. On the **Backups** page, locate the backup on the **Replica Sets** tab and click **Restore** in the **Operation** column.

Figure 9-14 Restoring a replica set instance backup



Step 5 In the **Restore DB Instance** dialog box, select **Restore to Original** for **Restoration Method** and click **OK**.

Figure 9-15 Restore to Original



- On the **Instances** page, the status of the instance changes from **Restoring** to **Available**.
- After the restoration is complete, a full backup will be automatically triggered.

----End

9.3.3 Restoring a Single Node Backup to the Original Instance

Scenarios

DDS allows you to restore an existing automated or manual backup to an original instance. The restored data is the same as the backup data.

When you restore an instance from a backup file, a full backup file is downloaded from OBS and then restored to the instance at an average speed of 40 MB/s.

NOTE

Huawei Cloud has discontinued the sale of DDS single node instances since July 15, 2023.

Precautions

- Restoring backup data to the original instance will overwrite existing data on the instance and cause the instance to be unavailable during the restoration. Exercise caution when performing this operation.
- The administrator password of the instance remains unchanged after the restoration.
- If you restore a manual backup, check whether the instance to which the manual backup belongs exists. If the instance does not exist, the backup can only be restored to a new instance.

Procedure

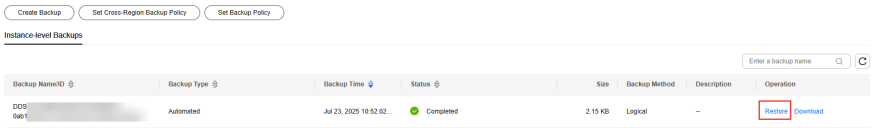
Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

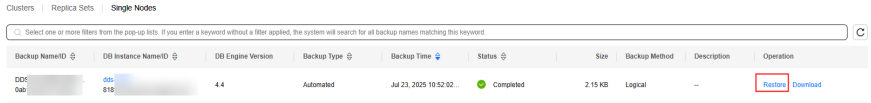
Step 4 On the **Instances** page, click the single node instance name. Choose **Backups & Restorations** in the navigation pane on the left, select the backup to be restored, and click **Restore**.

Figure 9-16 Restoring a single node backup



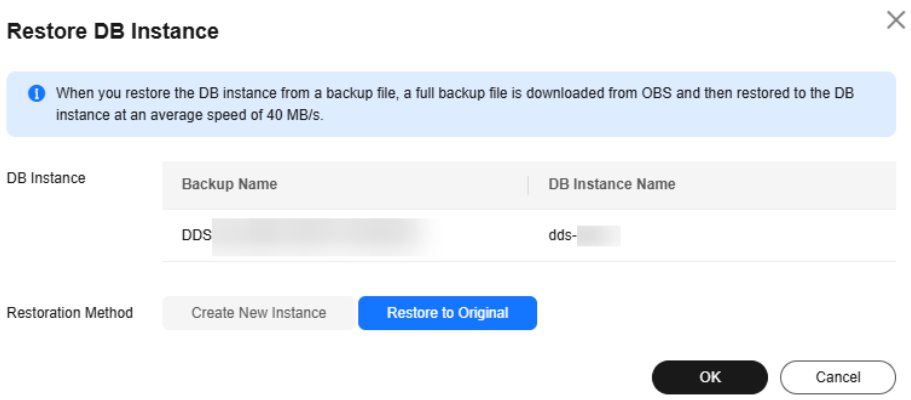
Alternatively, on the navigation pane on the left, choose **Backups**. On the **Backups** page, locate the target backup on the **Single Nodes** tab and click **Restore** in the **Operation** column.

Figure 9-17 Restoring a single node backup



Step 5 In the **Restore DB Instance** dialog box, select **Restore to Original** for **Restoration Method** and click **OK**.

Figure 9-18 Restore to Original



- On the **Instances** page, the status of the instance changes from **Restoring** to **Available**.
- After the restoration is complete, a full backup will be automatically triggered.

----End

9.4 Restoring Data to a Point in Time

9.4.1 Restoring a Cluster Instance to a Point in Time

Scenarios

DDS allows you to restore a cluster instance from a backup file to a point in time.

When you enter the point in time that you want to restore the instance to, DDS downloads the most recent full backup file from OBS to the instance. Then, incremental backups are also restored to the specified point in time on the instance. Data is restored at an average speed of 30 MB/s.

Precautions

- Only cluster instances of version 4.0 or later can be restored to a specified point in time.
- Data can be restored to a specific point in time only after the automated and incremental backup policies are enabled.
- Data can be restored to a new instance or the original instance.
- To ensure data security, the dropDatabase operation is blocked when the incremental backup is restored to a point in time. Empty databases or views may exist after the restoration. You can delete them.
- Data cannot be restored to a point in time in any of the following scenarios: rename operation, collmod operation, creating a user, deleting a user, creating

a role, deleting a role, enabling shard IP addresses of a cluster instance, changing the password of the shard node user, enabling config IP addresses of a cluster instance, changing the password of the config node user, and changing the password of the **rwuser** user. When a restricted scenario occurs, the incremental backup stops. After the next automated full backup, the incremental backup resumes.

- If the time window of the full backup overlaps with that of the incremental backup, the full backup prefers. The incremental backup is restricted so that a few time ranges are not within the recovery time window.
- If the incremental oplog traffic is greater than 250 GB/h or 75 MB/s, the incremental backup speed may not keep up with the oplog generation speed. As a result, some restoration time points may be unavailable.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the cluster instance name.

Step 5 In the navigation pane on the left, choose **Backups & Restorations**.

Step 6 On the **Backups & Restorations** page, click **Restore to Point in Time**.

Step 7 Select the date and time range, select or enter a time point within the acceptable range, and select **Create New Instance** or **Restore to Original**.

Step 8 On the displayed page, the instance is restored based on the restoration method you selected in **Step 7**.

- Create New Instance

  The **Create New Instance** page is displayed for you to create an instance using the backup data. The new DB instance is independent of the original one.

  - You are recommended to deploy the restored instance in a different AZ to ensure that applications will not be adversely affected by the failure in any single AZ.
  - The database type, DB instance type, compatible MongoDB version, storage engine, and storage type must be the same as those of the original and cannot be changed.
  - The storage space is the same as that of the original instance by default. You can increase the storage space, but you cannot reduce it.
  - Other settings can be modified. For details, see **Buying a Cluster Instance**.

- Restore to Original

  Check that the status of the instance on the **Instances** page is **Restoring**.

NOTICE

- Restoring backup data to the original instance will overwrite existing data on the instance and cause the instance to be unavailable during the restoration. Exercise caution when performing this operation.
- The administrator password of the instance remains unchanged after the restoration.

----End

9.4.2 Restoring a Replica Set Instance to a Point in Time

Scenarios

You can restore a replica set instance from a backup file to a specific point in time.

When you enter the point in time that you want to restore the instance to, DDS downloads the most recent full backup file from OBS to the instance. Then, incremental backups are also restored to the specified point in time on the instance. Data is restored at an average speed of 30 MB/s.

Precautions

- Currently, you can restore a replica set instance to a new or original DB instance at a point in time.
- Only replica set instances of version 4.0 or later can be restored to a point in time.
- Data can be restored to a specific point in time only after the automated backup policy is enabled.
- The local database is not included in the databases that can be restored to a specified time point.
- To ensure data security, the dropDatabase operation is blocked when the incremental backup is restored to a point in time. Empty databases or views may exist after the restoration. You can delete them.
- If the incremental oplog traffic is greater than 250 GB/h, the incremental backup speed may not keep up with the oplog generation speed. As a result, some restoration time points may be unavailable.

NOTE

If the window for restoring to a database or table time point is still not displayed after this function is enabled by , check whether the version is the latest. Upgrade the kernel to the latest minor version and refresh the page to check whether the window is displayed.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

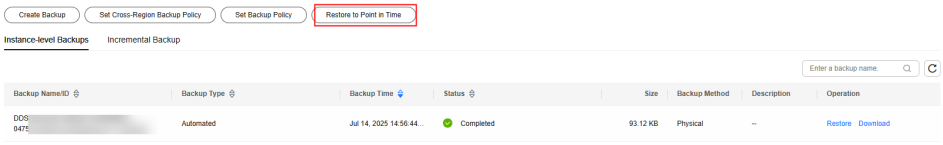
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the replica set instance name.

Step 5 In the navigation pane on the left, choose **Backups & Restorations**.

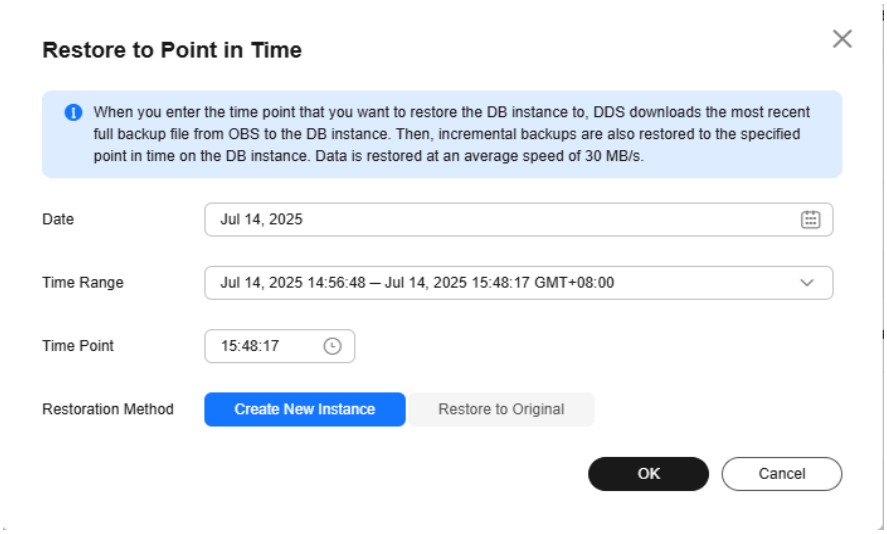
Step 6 On the **Backups & Restorations** page, click **Restore to Point in Time**.

Figure 9-19 Restoring to a point in time



Step 7 Select the date and time range, select or enter a time point within the acceptable range, and select **Create New Instance** or **Restore to Original**.

Figure 9-20 Restoring to a point in time



Step 8 On the displayed page, the DB instance is restored based on the restoration method you selected in **Step 7**.

- Create New Instance

  The **Create New Instance** page is displayed for you to create an instance using the backup data. The new DB instance is independent of the original one.

  - You are recommended to deploy the restored DB instance in a different AZ to ensure that applications will not be adversely affected by the failure in any single AZ.
  - The database type, DB instance type, compatible MongoDB version, storage engine, and storage type must be the same as those of the original and cannot be changed.
  - The storage space is the same as that of the original instance by default. You can increase the storage space, but you cannot reduce it.
  - Other settings have default values and can be modified. For details, see **Buying a Replica Set Instance**.

- Restore to Original

NOTICE

- Restoring backup data to the original instance will overwrite existing data on the instance and cause the instance to be unavailable during the restoration. Exercise caution when performing this operation.
- The administrator password of the instance remains unchanged after the restoration.
- If the backup method is logical backup, the backup cannot be restored to the original instance.

Check that the status of the DB instance on the **Instances** page is **Restoring**.

----End

9.4.3 Restoring a Replica Set Database or Table to a Specific Point in Time

Scenarios

DDS allows you to restore databases and tables using point-in-time recovery (PITR). This ensures your data integrity and minimizes impact on the original instance performance. You can select databases or tables and restore them to a specified point in time. During database or table PITR, the system restores the full and incremental data at the selected time point to a temporary instance, automatically exports the databases and tables to be restored, and then restores the databases and tables to the original instance. The time required depends on the amount of data to be backed up and restored on the instance. Please wait.

Restoring databases and tables will not overwrite data in the instance. You can select databases and tables to be restored.

Precautions

- Currently, only replica set instances of version 4.0 support the point-in-time recovery at the database and table level.
- Before performing the restoration, you need to enable the incremental backup policy.
- After a successful restoration, a new table named ***original-table-name*_bak_*timestamp*** is generated in the instance by default. If the table contains an index, the namespace of the index is changed to ***original-database-name.original-table-name*_bak_*timestamp***. You can rename the table later as required.
- New databases and tables will be generated in the original DB instance. Ensure that sufficient storage space is available.
- The length of *<Database name>.<Table name>* cannot exceed 120 characters. The length of *<Database name>.<Table name>.<Index name>* cannot exceed 128 characters, or the restoration may fail.
- Ensure that the name of the restored table is different from that of the existing table, or the restoration may fail.

- If you perform a table-level restoration and the table does not exist at the required point in time, an empty table is automatically created. If you perform a database-level restoration, the missing table is not created.
- If the incremental oplog traffic is greater than 250 GB/h, the incremental backup speed may not keep up with the oplog generation speed. As a result, some restoration time points may be unavailable.

Restrictions

- Database- and table-level restoration is related to CPU and memory specifications. For details about the maximum size of a single data record that can be restored to a specified time point, see **Table 9-2**.
- If a database- and table-level restoration fails, you can upgrade the specifications or batch restore the databases and tables to a specified time point.

Table 9-2 Specifications

CPU Type	Specifications	vCPUs	Memory (GB)	Maximum Size of a Single Data Record That Can Be Restored in a Collection
x86	General-purpose	2	4	400 KB
		2	8	800 KB
		4	8	1 MB
		4	16	1.3 MB
		8	16	1.3 MB
		8	32	2 MB
	Enhanced II	1	8	400 KB
		2	8	800 KB
		2	16	800 KB
		4	16	1.3 MB
		4	32	1.3 MB
		8	32	2 MB
		8	64	3 MB
		16	64	4 MB
		16	128	7 MB
		32	128	7 MB
		32	256	10 MB

CPU Type	Specifications	vCPUs	Memory (GB)	Maximum Size of a Single Data Record That Can Be Restored in a Collection
		64	256	10 MB
		64	512	16 MB
Kunpeng	-	2	4	400 KB
	-	2	8	800 KB
	-	4	8	1 MB
	-	4	16	1.3 MB
	-	8	16	1.3 MB
	-	8	32	2 MB
	-	16	32	2 MB
	-	16	64	4 MB

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the replica set instance.

Step 5 In the navigation pane on the left, choose **Backups & Restorations**.

Step 6 On the **Backups & Restorations** page, click **Restore Database and Table**.

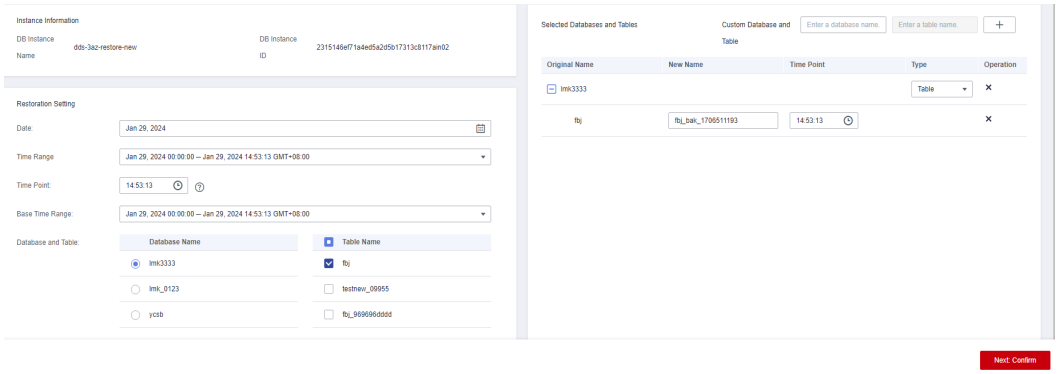
Step 7 In the displayed dialog box, configure parameters as required. The data in the new database and table is the same as that in the database and table at the selected time point.

Table 9-3 Database information

Parameter	Description
Date	Date when the automated backup of the DB instance is generated.
Time Range	Time range during which the automated backup can be restored.

Parameter	Description
Time Point	The specific point in time when the automated full backup is generated.
Base Time Range	Time range during which the database and table can be restored based on the automated full backup.
Database and Table	Databases and tables that have been automatically backed up within the base time range are displayed on the left. Select the databases and tables on the left to sync information to the area on the right.
Time Point	The point in time within the base time range.
Custom Database and Table	You can add custom databases and tables as required. • The system databases cannot be restored. Therefore, the database name cannot be **admin**, **local**, or **config**. • The database name cannot contain spaces and the following special characters: ".$\/*?~#:| • A table name cannot use "system" as the prefix. • The length of *<Database name>.<Table name>* cannot exceed 120 characters. The length of *<Database name>.<Table name>.<Index name>* cannot exceed 128 characters, or the restoration may fail. • Ensure that the name of the restored table is different from that of the existing table. Otherwise, the restoration may fail. • After a successful restoration, a new table named ***original-table-name*_bak_*timestamp*** is generated in the instance by default. If the table contains an index, the namespace of the index is changed to ***original-database-name.original-table-name*_bak_*timestamp***. You can rename the table later as required. To distinguish the point in time of the custom databases and tables from those synchronized on the right, set the point in time to a different value. The system restores data to the custom databases and tables based on the time configured here.
Type	You can restore data to a database or table. If you perform a table-level restoration and the table does not exist at the required point in time, an empty table is automatically created. If you perform a database-level restore, data will be restored to the database separately, and the table will not be created.

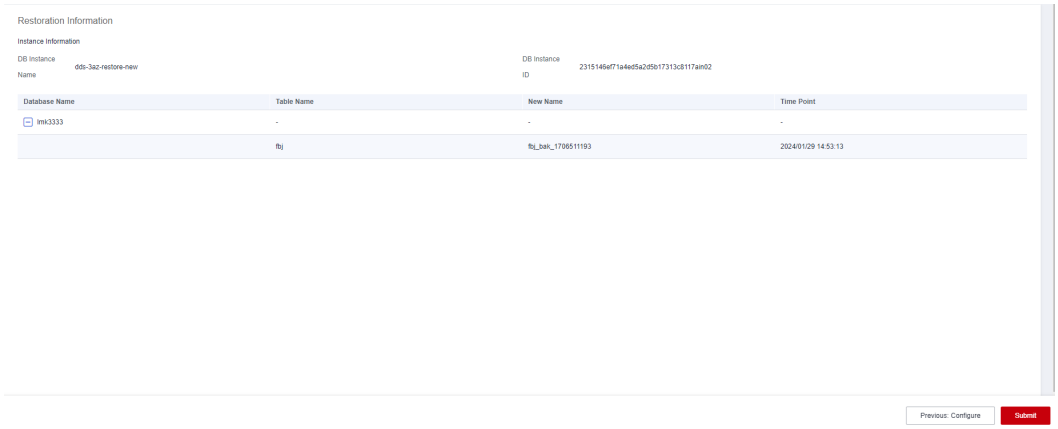
Figure 9-21 Selecting database and table



Step 8 Click **Next: Confirm**.

Step 9 Click **Submit** to start the restoration.

Figure 9-22 Confirming the information



Step 10 On the **Instances** page, the DB instance status is **Restoring**. During the restoration process, services are not interrupted.

Step 11 After the restoration is successful, manage data in the database and table as required.

If you need to use the original database and table names, you can use a rename operation to back up the original database and table and switch your service to the restored database and table. Then, delete the original database and table after ensuring that your services are normal.

Example:

db.adminCommand({renameCollection: "db1.test1", to: "db2.test2"})

The above command is used to move the **test1** table from the **db1** database to the **db2** database and rename the table to **test2**.

----End

9.4.4 Restoring a Cluster Database or Table to a Specific Point in Time

Scenarios

DDS allows you to restore databases and tables using point-in-time recovery (PITR). This ensures your data integrity and minimizes impact on the original instance performance. You can select databases or tables and restore them to a specified point in time. During database or table PITR, the system restores the full and incremental data at the selected time point to a temporary instance, automatically exports the databases and tables to be restored, and then restores the databases and tables to the original instance. The time required depends on the amount of data to be backed up and restored on the instance. Please wait.

Restoring databases and tables will not overwrite data in the instance. You can select databases and tables to be restored.

Precautions

- Currently, only cluster instances of version 4.0 support the point-in-time recovery at the database and table level.
- Before performing the restoration, you need to enable the incremental backup policy.
- After a successful restoration, a new table named ***original-table-name*_bak_*timestamp*** is generated in the instance by default. If the table contains an index, the namespace of the index is changed to ***original-database-name.original-table-name*_bak_*timestamp***. You can rename the table later as required.
- New databases and tables will be generated in the original DB instance. Ensure that sufficient storage space is available.
- The length of *<Database name>.<Table name>* cannot exceed 120 characters. The length of *<Database name>.<Table name>.<Index name>* cannot exceed 128 characters, or the restoration may fail.
- Ensure that the name of the restored table is different from that of the existing table, or the restoration may fail.
- If the time window of the full backup overlaps with that of the incremental backup, the full backup prefers. The incremental backup is restricted so that a few time ranges are not within the recovery time window.
- If you perform a table-level restoration and the table does not exist at the required point in time, an empty table is automatically created. If you perform a database-level restoration, the missing table is not created.
- If the incremental oplog traffic is greater than 250 GB/h, the incremental backup speed may not keep up with the oplog generation speed. As a result, some restoration time points may be unavailable.

Restrictions

- Database- and table-level restoration is related to CPU and memory specifications. For details about the maximum size of a single data record that can be restored to a specified time point, see **Table 9-4**.

- If a database- and table-level restoration fails, you can upgrade the specifications or batch restore the databases and tables to a specified time point.

Table 9-4 Specifications

CPU Type	Specifications	vCPUs	Memory (GB)	Maximum Size of a Single Data Record That Can Be Restored in a Collection
x86	General-purpose	2	4	400 KB
		2	8	800 KB
		4	8	1 MB
		4	16	1.3 MB
		8	16	1.3 MB
		8	32	2 MB
	Enhanced II	1	8	400 KB
		2	8	800 KB
		2	16	800 KB
		4	16	1.3 MB
		4	32	1.3 MB
		8	32	2 MB
		8	64	3 MB
		16	64	4 MB
		16	128	7 MB
		32	128	7 MB
		32	256	10 MB
		64	256	10 MB
		64	512	16 MB
Kunpeng	-	2	4	400 KB
	-	2	8	800 KB
	-	4	8	1 MB
	-	4	16	1.3 MB
	-	8	16	1.3 MB

CPU Type	Specifications	vCPUs	Memory (GB)	Maximum Size of a Single Data Record That Can Be Restored in a Collection
	-	8	32	2 MB
	-	16	32	2 MB
	-	16	64	4 MB

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the target cluster instance and click its name.

Step 5 In the navigation pane on the left, choose **Backups & Restorations**.

Step 6 On the **Backups & Restorations** page, click **Restore Database and Table**.

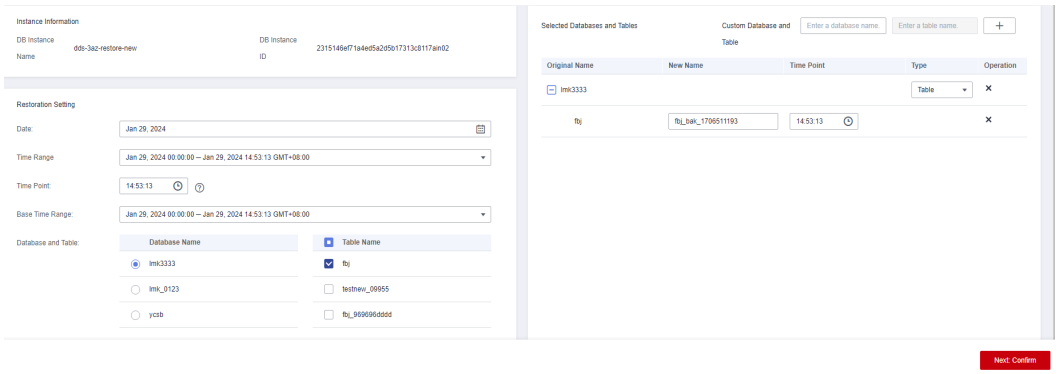
Step 7 In the displayed dialog box, configure parameters as required. The data in the new database and table is the same as that in the database and table at the selected time point.

Table 9-5 Database information

Parameter	Description
Date	Date when the automated backup of the DB instance is generated.
Time Range	Time range during which the automated backup can be restored.
Time Point	The specific point in time when the automated full backup is generated.
Base Time Range	Time range during which the database and table can be restored based on the automated full backup.
Database and Table	Databases and tables that have been automatically backed up within the base time range are displayed on the left. Select the databases and tables on the left to sync information to the area on the right.
Time Point	The point in time within the base time range.

Parameter	Description
Custom Database and Table	You can add custom databases and tables as required. ● The system databases cannot be restored. Therefore, the database name cannot be **admin**, **local**, or **config**. ● The database name cannot contain spaces and the following special characters: ".$\/*?~#:| ● A table name cannot use "system" as the prefix. ● The length of *<Database name>.<Table name>* cannot exceed 120 characters. The length of *<Database name>.<Table name>.<Index name>* cannot exceed 128 characters, or the restoration may fail. ● Ensure that the name of the restored table is different from that of the existing table. Otherwise, the restoration may fail. ● After a successful restoration, a new table named ***original-table-name*_bak_*timestamp*** is generated in the instance by default. If the table contains an index, the namespace of the index is changed to ***original-database-name.original-table-name*_bak_*timestamp***. You can rename the table later as required. To distinguish the point in time of the custom databases and tables from those synchronized on the right, set the point in time to a different value. The system restores data to the custom databases and tables based on the time configured here.
Type	You can restore data to a database or table. If you perform a table-level restoration and the table does not exist at the required point in time, an empty table is automatically created. If you perform a database-level restoration and the database does not exist at the required point in time, an empty database is automatically created. If you perform a database-level restoration, a new database is created. The table name in the new database is the same as that in the database to be restored.

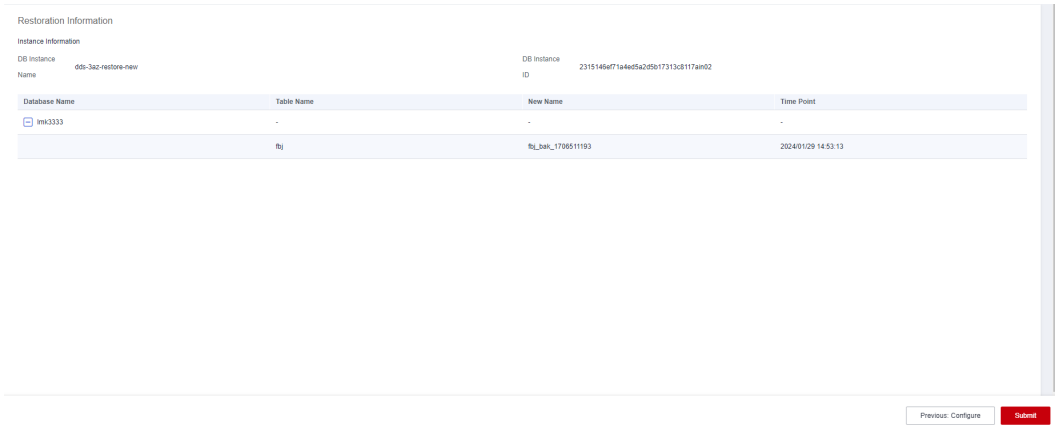
Figure 9-23 Database and Table



Step 8 Click **Next: Confirm**.

Step 9 Click **Submit** to start the restoration.

Figure 9-24 Confirming the information



Step 10 On the **Instances** page, the DB instance status is **Restoring**. During the restoration process, services are not interrupted.

Step 11 After the restoration is successful, manage data in the database and table as required.

If you need to use the original database and table names, you can use a rename operation to back up the original database and table and switch your service to the restored database and table. Then, delete the original database and table after ensuring that your services are normal.

Example:

db.adminCommand({renameCollection: "db1.test1", to: "db2.test2"})

The above command is used to move the **test1** table from the **db1** database to the **db2** database and rename the table to **test2**.

----End

9.5 Restoring Data to an On-Premises Database

9.5.1 Restoring a Cluster Backup to an On-premises Database

9.5.1.1 Overview

Scenarios

This section uses the Linux operating system as an example to describe how to restore the downloaded backup file of a cluster instance to your on-premises database. For details about how to download backup files, see **Downloading a Backup File**.

Precautions

- This method applies only to cluster instances.
- Only DDS 3.4 and 4.0 instances can be restored in this method. DDS 4.2 or later does not support this method.
- The directories, IP addresses, and ports provided in the example are for reference only. Configure these items based on your service requirements.
- There is one backup file of the configsvr node and multiple backup files of the shardsrv node. The number of backup files depends on the number of shardsvr nodes.
- After the backup file is downloaded, decompress the file using LZ4. Command for reference: **lz4 -d $1 | tar -xC $2**

  $1: indicates the downloaded backup file.

  $2: indicates the directory to which the backup file is decompressed.
- For details about how to migrate data at the database or collection level, see **Migrating Data Using mongodump and mongorestore**.

Prerequisites

MongoDB client 3.4 or 4.0 has been installed on your on-premises database.

9.5.1.2 Directories and Configurations

NOTICE

The local directory, configuration file, and configuration information are not fixed and can be customized.

The following uses backup files of two shardsvr cluster instances as an example (instance ID: cac1efc8e65e42ecad8953352321bfeein02).

- Directory of the decompressed backup files of the configsvr node: /compile/download/backups/cac1efc8e65e42ecad8953352321bfeein02_41c8a32fb10245899708dea453a8c5c9no02
- Directory of the decompressed backup files of the shardsvr1 node:

/compile/download/backups/cac1efc8e65e42ecad8953352321bfeein02_6cfa6167d4114d7c8cec5b47f9a78dc5no02

- Directory of the decompressed backup files of the shardsvr2 node:

  /compile/download/backups/cac1efc8e65e42ecad8953352321bfeein02_92b196d2401041a7af869a2a3cab7079no02

Data directories and log directories of the three configsvr nodes

/compile/cluster-restore/cfg1/data/db

/compile/cluster-restore/cfg1/log

/compile/cluster-restore/cfg2/data/db

/compile/cluster-restore/cfg2/log

/compile/cluster-restore/cfg3/data/db

/compile/cluster-restore/cfg3/log

Data directories and log directories of the three nodes of shardsvr1

/compile/cluster-restore/shd11/data/db

/compile/cluster-restore/shd11/log

/compile/cluster-restore/shd12/data/db

/compile/cluster-restore/shd12/log

/compile/cluster-restore/shd13/data/db

/compile/cluster-restore/shd13/log

Data directories and log directories of the three nodes of shardsvr2

/compile/cluster-restore/shd21/data/db

/compile/cluster-restore/shd21/log

/compile/cluster-restore/shd22/data/db

/compile/cluster-restore/shd22/log

/compile/cluster-restore/shd23/data/db

/compile/cluster-restore/shd23/log

Log directories of the dds mongos node

/compile/cluster-restore/mgs1/log

/compile/cluster-restore/mgs2/log

IP Address and Port Information

The IP address bound to the process is 127.0.0.1. The port numbers are allocated as follows:

- dds mongos node: 40301, 40302
- configsvr node: 40303, 40304, 40305
- shardsvr1: 40306, 40307, and 40308
- shardsvr2: 40309, 40310, and 40311

Configuration file description

- Configuration file of a single node and configuration files of three nodes in the configsvr replica set

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/single_40303.yaml

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/configsvr_40303.yaml

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/configsvr_40304.yaml

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/configsvr_40305.yaml
- Configuration file of a single node and configuration files of three nodes in the shardsvr1 replica set

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/single_40306.yaml

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/shardsvr_40306.yaml

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/shardsvr_40307.yaml

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/shardsvr_40308.yaml
- Configuration file of a single node and configuration files of three nodes in the shardsvr2 replica set:

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/single_40309.yaml

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/shardsvr_40309.yaml

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/shardsvr_40310.yaml

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/shardsvr_40311.yaml
- Configuration file of the dds mongos node:

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/mongos_40301.yaml

  /compile/mongodb/mongodb-src-4.0.3/restoreconfig/mongos_40302.yaml

Procedure

Command running directory: /compile/mongodb/mongodb-src-4.0.3

9.5.1.3 Restoring the configsvr Replica Set

Preparing Directories

rm -rf /compile/cluster-restore/cfg*

mkdir -p /compile/cluster-restore/cfg1/data/db

mkdir -p /compile/cluster-restore/cfg1/log

mkdir -p /compile/cluster-restore/cfg2/data/db

mkdir -p /compile/cluster-restore/cfg2/log

mkdir -p /compile/cluster-restore/cfg3/data/db

mkdir -p /compile/cluster-restore/cfg3/log

Procedure

Step 1 Prepare the configuration file and data directory of a single node and start the process in single-node mode.

1. The configuration file is as follows (restoreconfig/single_40303.yaml):

```
net:
  bindIp: 127.0.0.1
  port: 40303
  unixDomainSocket: {enabled: false}
processManagement: {fork: true, pidFilePath: /compile/cluster-restore/cfg1/configsvr.pid}
storage:
  dbPath: /compile/cluster-restore/cfg1/data/db/
  directoryPerDB: true
  engine: wiredTiger
  wiredTiger:
    collectionConfig: {blockCompressor: snappy}
    engineConfig: {directoryForIndexes: true, journalCompressor: snappy}
    indexConfig: {prefixCompression: true}
systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/cfg1/log/configsingle.log}
```

2. Copy the decompressed **configsvr** file to the **dbPath** directory on the single node.

   cp -aR

   /compile/download/backups/cac1efc8e65e42ecad8953352321bfeein02_41c8a32fb10245899708dea453a8c5c9no02/* /compile/cluster-restore/cfg1/data/db/

3. Start the process.

   ./mongod -f restoreconfig/single_40303.yaml

Step 2 Connect to the single node and run the following configuration command:

./mongo --host 127.0.0.1 --port 40303

1. Run the following commands to modify the replica set configuration:

   var cf=db.getSiblingDB('local').system.replset.findOne();

   cf['members'][0]['host']='127.0.0.1:40303';

   cf['members'][1]['host']='127.0.0.1:40304';

   cf['members'][2]['host']='127.0.0.1:40305';

   cf['members'][0]['hidden']=false;

   cf['members'][1]['hidden']=false;

   cf['members'][2]['hidden']=false;

   cf['members'][0]['priority']=1;

   cf['members'][1]['priority']=1;

   cf['members'][2]['priority']=1;

   db.getSiblingDB('local').system.replset.remove({});

   db.getSiblingDB('local').system.replset.insert(cf)

2. Run the following commands to clear the built-in accounts:

   db.getSiblingDB('admin').dropAllUsers();

   db.getSiblingDB('admin').dropAllRoles();

3. Run the following command to update the dds mongos and shard information:

   db.getSiblingDB('config').mongos.remove({});

   Query the _id information about multiple shards in the **config.shards** table. The _id information is used as the query condition of _id in the following statements. Update records in sequence.

   db.getSiblingDB('config').shards.update({'_id' : 'shard_1'},{$set: {'host': 'shard_1/127.0.0.1:40306,127.0.0.1:40307,127.0.0.1:40308'}})

   db.getSiblingDB('config').shards.update({'_id' : 'shard_2'},{$set: {'host': 'shard_2/127.0.0.1:40309,127.0.0.1:40310,127.0.0.1:40311'}})

   db.getSiblingDB('config').mongos.find({});

   db.getSiblingDB('config').shards.find({});

4. Run the following command to stop the single-node process:

   db.getSiblingDB('admin').shutdownServer();

Step 3 Create a configsvr replica set.

1. Copy the **dbPath** file of the configsvr1 node to the directories of the other two configsvr nodes.

   cp -aR /compile/cluster-restore/cfg1/data/db/ /compile/cluster-restore/cfg2/data/db/

   cp -aR /compile/cluster-restore/cfg1/data/db/ /compile/cluster-restore/cfg3/data/db/

2. Add the replica set configuration attribute to the configuration file (**restoreconfig/configsvr_40303.yaml**) of the configsvr-1 node.

```
net:
  bindIp: 127.0.0.1
  port: 40303
  unixDomainSocket: {enabled: false}
processManagement: {fork: true, pidFilePath: /compile/cluster-restore/cfg1/configsvr.pid}
replication: {replSetName: config}
sharding: {archiveMovedChunks: false, clusterRole: configsvr}
storage:
  dbPath: /compile/cluster-restore/cfg1/data/db/
  directoryPerDB: true
  engine: wiredTiger
  wiredTiger:
    collectionConfig: {blockCompressor: snappy}
    engineConfig: {directoryForIndexes: true, journalCompressor: snappy}
    indexConfig: {prefixCompression: true}
systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/cfg1/log/configsvr.log}
```

3. Start the process.

   ./mongod -f restoreconfig/configsvr_40303.yaml

4. Add the replica set configuration attribute to the configuration file (**restoreconfig/configsvr_40304.yaml**) of the configsvr-2 node.

```
net:
  bindIp: 127.0.0.1
```



```
  port: 40304
  unixDomainSocket: {enabled: false}
processManagement: {fork: true, pidFilePath: /compile/cluster-restore/cfg2/configsvr.pid}
replication: {replSetName: config}
sharding: {archiveMovedChunks: false, clusterRole: configsvr}
storage:
  dbPath: /compile/cluster-restore/cfg2/data/db/
  directoryPerDB: true
  engine: wiredTiger
  wiredTiger:
    collectionConfig: {blockCompressor: snappy}
    engineConfig: {directoryForIndexes: true, journalCompressor: snappy}
    indexConfig: {prefixCompression: true}
systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/cfg2/log/configsvr.log}
```

5. Start the process.

   ./mongod -f restoreconfig/configsvr_40304.yaml

6. Add the replica set configuration attribute to the configuration file (**restoreconfig/configsvr_40305.yaml**) of the configsvr-3 node.

```
net:
  bindIp: 127.0.0.1
  port: 40305
  unixDomainSocket: {enabled: false}
processManagement: {fork: true, pidFilePath: /compile/cluster-restore/cfg3/configsvr.pid}
replication: {replSetName: config}
sharding: {archiveMovedChunks: false, clusterRole: configsvr}
storage:
  dbPath: /compile/cluster-restore/cfg3/data/db/
  directoryPerDB: true
  engine: wiredTiger
  wiredTiger:
    collectionConfig: {blockCompressor: snappy}
    engineConfig: {directoryForIndexes: true, journalCompressor: snappy}
    indexConfig: {prefixCompression: true}
systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/cfg3/log/configsvr.log}
```

7. Start the process.

   ./mongod -f restoreconfig/configsvr_40305.yaml

Step 4 Wait until the primary node is selected.

./mongo --host 127.0.0.1 --port 40303

Run the **rs.status()** command to check whether the primary node exists.

----End

9.5.1.4 Restoring the shardsvr1 Replica Set

Preparing Directories

rm -rf /compile/cluster-restore/shd1*

mkdir -p /compile/cluster-restore/shd11/data/db

mkdir -p /compile/cluster-restore/shd11/log

mkdir -p /compile/cluster-restore/shd12/data/db

mkdir -p /compile/cluster-restore/shd12/log

mkdir -p /compile/cluster-restore/shd13/data/db

mkdir -p /compile/cluster-restore/shd13/log

Procedure

Step 1 Prepare the configuration file and directory of a single node and start the process in single-node mode.

1. The configuration file is as follows (**restoreconfig/single_40306.yaml**):

```
net:
  bindIp: 127.0.0.1
  port: 40306
  unixDomainSocket: {enabled: false}
processManagement: {fork: true, pidFilePath: /compile/cluster-restore/shd11/mongod.pid}
storage:
  dbPath: /compile/cluster-restore/shd11/data/db/
  directoryPerDB: true
  engine: wiredTiger
  wiredTiger:
    collectionConfig: {blockCompressor: snappy}
    engineConfig: {directoryForIndexes: true, journalCompressor: snappy}
    indexConfig: {prefixCompression: true}
systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/shd11/log/mongod.log}
```

2. Copy the decompressed **shardsvr1** file to the **dbPath** directory on the single node.

   cp -aR

   /compile/download/backups/cac1efc8e65e42ecad8953352321bfeein02_6cfa6167d4114d7c8cec5b47f9a78dc5no02/* /compile/cluster-restore/shd11/data/db/

3. Start the process.

   ./mongod -f restoreconfig/single_40306.yaml

Step 2 Connect to the single node and run the following configuration command:

Connection command: ./mongo --host 127.0.0.1 --port 40306

1. Run the following commands to modify the replica set configuration:

   var cf=db.getSiblingDB('local').system.replset.findOne();

   cf['members'][0]['host']='127.0.0.1:40306';

   cf['members'][1]['host']='127.0.0.1:40307';

   cf['members'][2]['host']='127.0.0.1:40308';

   cf['members'][0]['hidden']=false;

   cf['members'][1]['hidden']=false;

   cf['members'][2]['hidden']=false;

   cf['members'][0]['priority']=1;

   cf['members'][1]['priority']=1;

   cf['members'][2]['priority']=1;

   db.getSiblingDB('local').system.replset.remove({});

db.getSiblingDB('local').system.replset.insert(cf)

2. Run the following commands to clear the built-in accounts:

   db.getSiblingDB('admin').dropAllUsers();

   db.getSiblingDB('admin').dropAllRoles();

3. Run the following commands to update the configsvr information:

   Connection command: ./mongo --host 127.0.0.1 --port 40306

   var vs = db.getSiblingDB('admin').system.version.find();

   while (vs.hasNext()) {

   var curr = vs.next();

   if (curr.hasOwnProperty('configsvrConnectionString')) {

   db.getSiblingDB('admin').system.version.update({'_id' : curr._id}, {$set: {'configsvrConnectionString': 'config/127.0.0.1:40303,127.0.0.1:40304,127.0.0.1:40305'}});

   }

   }

4. Run the following command to stop the single-node process:

   db.getSiblingDB('admin').shutdownServer();

Step 3 Create the shardsvr1 replica set.

1. Copy the **dbPath** file of the shardsvr1 node to the directories of the other two shardsvr nodes.

   cp -aR /compile/cluster-restore/shd11/data/db/ /compile/cluster-restore/shd12/data/db/

   cp -aR /compile/cluster-restore/shd11/data/db/ /compile/cluster-restore/shd13/data/db/

2. Add the replica set configuration attribute to the configuration file (**restoreconfig/shardsvr_40306.yaml**) of the shardsvr1-1 node.

   --- For details about the value of **replication.replSetName**, see the shard _id information in **Step 2.3**.

```
net:
  bindIp: 127.0.0.1
  port: 40306
  unixDomainSocket: {enabled: false}
processManagement: {fork: true, pidFilePath: /compile/cluster-restore/shd11/mongod.pid}
replication: {replSetName: shard_1}
sharding: {archiveMovedChunks: false, clusterRole: shardsvr}
storage:
  dbPath: /compile/cluster-restore/shd11/data/db/
  directoryPerDB: true
  engine: wiredTiger
  wiredTiger:
    collectionConfig: {blockCompressor: snappy}
    engineConfig: {directoryForIndexes: true, journalCompressor: snappy}
    indexConfig: {prefixCompression: true}
systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/shd11/log/mongod.log}
```

3. Start the process.

   ./mongod -f restoreconfig/shardsvr_40306.yaml

4. Add the replica set configuration attribute to the configuration file (**restoreconfig/shardsvr_40307.yaml**) of the shardsvr1-2 node.

   --- For details about the value of **replication.replSetName**, see the shard _id information in **Step 2.3**.

   ```
   net:
     bindIp: 127.0.0.1
     port: 40307
     unixDomainSocket: {enabled: false}
   processManagement: {fork: true, pidFilePath: /compile/cluster-restore/shd12/mongod.pid}
   replication: {replSetName: shard_1}
   sharding: {archiveMovedChunks: false, clusterRole: shardsvr}
   storage:
     dbPath: /compile/cluster-restore/shd12/data/db/
     directoryPerDB: true
     engine: wiredTiger
     wiredTiger:
       collectionConfig: {blockCompressor: snappy}
       engineConfig: {directoryForIndexes: true, journalCompressor: snappy}
       indexConfig: {prefixCompression: true}
   systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/shd12/log/mongod.log}
   ```

5. Start the process.

   ./mongod -f restoreconfig/shardsvr_40307.yaml

6. Add the replica set configuration attribute to the configuration file (**restoreconfig/shardsvr_40308.yaml**) of the shardsvr1-3 node.

   --- For details about the value of **replication.replSetName**, see the shard _id information in **Step 2.3**.

   ```
   net:
     bindIp: 127.0.0.1
     port: 40308
     unixDomainSocket: {enabled: false}
   processManagement: {fork: true, pidFilePath: /compile/cluster-restore/shd13/mongod.pid}
   replication: {replSetName: shard_1}
   sharding: {archiveMovedChunks: false, clusterRole: shardsvr}
   storage:
     dbPath: /compile/cluster-restore/shd13/data/db/
     directoryPerDB: true
     engine: wiredTiger
     wiredTiger:
       collectionConfig: {blockCompressor: snappy}
       engineConfig: {directoryForIndexes: true, journalCompressor: snappy}
       indexConfig: {prefixCompression: true}
   systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/shd13/log/mongod.log}
   ```

7. Start the process.

   ./mongod -f restoreconfig/shardsvr_40308.yaml

Step 4 Wait until the primary node is selected.

./mongo --host 127.0.0.1 --port 40306

Run the **rs.status()** command to check whether the primary node exists.

----End

9.5.1.5 Restoring the shardsvr2 Replica Set

Preparing Directories

rm -rf /compile/cluster-restore/shd2*

mkdir -p /compile/cluster-restore/shd21/data/db

mkdir -p /compile/cluster-restore/shd21/log

mkdir -p /compile/cluster-restore/shd22/data/db

mkdir -p /compile/cluster-restore/shd22/log

mkdir -p /compile/cluster-restore/shd23/data/db

mkdir -p /compile/cluster-restore/shd23/log

Procedure

Step 1 Prepare the configuration file and directory of a single node and start the process in single-node mode.

1. The configuration file is as follows (**restoreconfig/single_40309.yaml**):

```
net:
  bindIp: 127.0.0.1
  port: 40309
  unixDomainSocket: {enabled: false}
processManagement: {fork: true, pidFilePath: /compile/cluster-restore/shd21/mongod.pid}
storage:
  dbPath: /compile/cluster-restore/shd21/data/db/
  directoryPerDB: true
  engine: wiredTiger
  wiredTiger:
    collectionConfig: {blockCompressor: snappy}
    engineConfig: {directoryForIndexes: true, journalCompressor: snappy}
    indexConfig: {prefixCompression: true}
systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/shd21/log/mongod.log}
```

1. Copy the decompressed **shardsvr2** file to the **dbPath** directory on the single node.

   cp -aR

   /compile/download/backups/cac1efc8e65e42ecad8953352321bfeein02_92b196d2401041a7af869a2a3cab7079no02/* /compile/cluster-restore/shd21/data/db/

2. Start the process.

   ./mongod -f restoreconfig/single_40309.yaml

Step 2 Connect to the single node and run the following configuration command:

Connection command: ./mongo --host 127.0.0.1 --port 40309

1. Run the following commands to modify the replica set configuration:

   var cf=db.getSiblingDB('local').system.replset.findOne();

   cf['members'][0]['host']='127.0.0.1:40309';

   cf['members'][1]['host']='127.0.0.1:40310';


```
cf['members'][2]['host']='127.0.0.1:40311';
cf['members'][0]['hidden']=false;
cf['members'][1]['hidden']=false;
cf['members'][2]['hidden']=false;
cf['members'][0]['priority']=1;
cf['members'][1]['priority']=1;
cf['members'][2]['priority']=1;
db.getSiblingDB('local').system.replset.remove({});
db.getSiblingDB('local').system.replset.insert(cf)
```

2. Run the following commands to clear the built-in accounts:

```
db.getSiblingDB('admin').dropAllUsers();
db.getSiblingDB('admin').dropAllRoles();
```

3. Run the following commands to update the configsvr information:

```
var vs = db.getSiblingDB('admin').system.version.find();
while (vs.hasNext()) {
var curr = vs.next();
if (curr.hasOwnProperty('configsvrConnectionString')) {
db.getSiblingDB('admin').system.version.update({'_id' : curr._id}, {$set:
{'configsvrConnectionString': 'config/
127.0.0.1:40303,127.0.0.1:40304,127.0.0.1:40305'}});
}
}
```

4. Run the following command to stop the single-node process:

```
db.getSiblingDB('admin').shutdownServer();
```

Step 3 Create the shardsvr2 replica set.

1. Copy the **dbPath** file of the shardsvr2 node to the directories of the other two shardsvr nodes.

```
cp -aR /compile/cluster-restore/shd21/data/db/ /compile/cluster-restore/
shd22/data/db/
```

```
cp -aR /compile/cluster-restore/shd21/data/db/ /compile/cluster-restore/
shd23/data/db/
```

2. Add the replica set configuration attribute to the configuration file (**restoreconfig/shardsvr_40309.yaml**) of the shardsvr2-1 node.

--- For details about the value of **replication.replSetName**, see the shard _id information in **Step 2.3**.

```
net:
  bindIp: 127.0.0.1
  port: 40309
  unixDomainSocket: {enabled: false}
processManagement: {fork: true, pidFilePath: /compile/cluster-restore/shd21/mongod.pid}
replication: {replSetName: shard_2}
sharding: {archiveMovedChunks: false, clusterRole: shardsvr}
storage:
  dbPath: /compile/cluster-restore/shd21/data/db/
  directoryPerDB: true
```



```
  engine: wiredTiger
  wiredTiger:
    collectionConfig: {blockCompressor: snappy}
    engineConfig: {directoryForIndexes: true, journalCompressor: snappy}
    indexConfig: {prefixCompression: true}
systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/shd21/log/mongod.log}
```

3. Start the process.

   ./mongod -f restoreconfig/shardsvr_40309.yaml

4. Add the replica set configuration attribute to the configuration file (**restoreconfig/shardsvr_40310.yaml**) of the shardsvr2-2 node.

   --- For details about the value of **replication.replSetName**, see the shard _id information in **Step 2.3**.

```
net:
  bindIp: 127.0.0.1
  port: 40310
  unixDomainSocket: {enabled: false}
processManagement: {fork: true, pidFilePath: /compile/cluster-restore/shd22/mongod.pid}
replication: {replSetName: shard_2}
sharding: {archiveMovedChunks: false, clusterRole: shardsvr}
storage:
  dbPath: /compile/cluster-restore/shd22/data/db/
  directoryPerDB: true
  engine: wiredTiger
  wiredTiger:
    collectionConfig: {blockCompressor: snappy}
    engineConfig: {directoryForIndexes: true, journalCompressor: snappy}
    indexConfig: {prefixCompression: true}
systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/shd22/log/mongod.log}
```

5. Start the process.

   ./mongod -f restoreconfig/shardsvr_40310.yaml

6. Add the replica set configuration attribute to the configuration file (**restoreconfig/shardsvr_40311.yaml**) of the shardsvr2-3 node.

   --- For details about the value of **replication.replSetName**, see the shard _id information in **Step 2.3**.

```
net:
  bindIp: 127.0.0.1
  port: 40311
  unixDomainSocket: {enabled: false}
processManagement: {fork: true, pidFilePath: /compile/cluster-restore/shd23/mongod.pid}
replication: {replSetName: shard_2}
sharding: {archiveMovedChunks: false, clusterRole: shardsvr}
storage:
  dbPath: /compile/cluster-restore/shd23/data/db/
  directoryPerDB: true
  engine: wiredTiger
  wiredTiger:
    collectionConfig: {blockCompressor: snappy}
    engineConfig: {directoryForIndexes: true, journalCompressor: snappy}
    indexConfig: {prefixCompression: true}
systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/shd23/log/mongod.log}
```

7. Start the process.

   ./mongod -f restoreconfig/shardsvr_40311.yaml

Step 4 Wait until the primary node is selected.

./mongo --host 127.0.0.1 --port 40309

Run the **rs.status()** command to check whether the primary node exists.

----End

9.5.1.6 Restoring the dds mongos Node

Procedure

Step 1 Prepare the configuration file and directory of the dds mongos node.

rm -rf /compile/cluster-restore/mgs*

mkdir -p /compile/cluster-restore/mgs1/log

mkdir -p /compile/cluster-restore/mgs2/log

Step 2 Configuration file (restoreconfig/mongos_40301.yaml)

```
net:
  bindIp: 127.0.0.1
  port: 40301
  unixDomainSocket: {enabled: false}
processManagement: {fork: true, pidFilePath: /compile/cluster-restore/mgs1/mongos.pid}
sharding: {configDB: 'config/127.0.0.1:40303,127.0.0.1:40304,127.0.0.1:40305'}
systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/
mgs1/log/mongos.log}
```

Step 3 Configuration file (restoreconfig/mongos_40302.yaml)

```
net:
  bindIp: 127.0.0.1
  port: 40302
  unixDomainSocket: {enabled: false}
processManagement: {fork: true, pidFilePath: /compile/cluster-restore/mgs2/mongos.pid}
sharding: {configDB: 'config/127.0.0.1:40303,127.0.0.1:40304,127.0.0.1:40305'}
systemLog: {destination: file, logAppend: true, logRotate: reopen, path: /compile/cluster-restore/
mgs2/log/mongos.log}
```

Step 4 Start the mongo node.

./mongos -f restoreconfig/mongos_40301.yaml

./mongos -f restoreconfig/mongos_40302.yaml

----End

Checking the Cluster Status

Connect to the cluster through dds mongos and check the data status.

./mongo --host 127.0.0.1 --port 40301

./mongo --host 127.0.0.1 --port 40302

9.5.1.7 Checking the Cluster Status

Connect to the cluster through dds mongos and check the data status.

./mongo --host 127.0.0.1 --port 40301

./mongo --host 127.0.0.1 --port 40302

9.5.2 Restoring a Replica Set Backup to an On-Premises Database

Scenarios

To restore a DB instance backup file to an on-premises database, you can only use databases on Linux.

This section uses the Linux operating system as an example to describe how to restore the downloaded backup file of a replica set instance to your on-premises database. For details about how to download backup files, see **Downloading a Backup File**.

Precautions

- MongoDB client 3.4 has been installed on your on-premises MongoDB database.
- Only DDS 3.4 and 4.0 instances can be restored in this method. DDS 4.2 or later does not support this method.
- For details about how to migrate data at the database or collection level, see **Migrating Data Using mongodump and mongorestore**.

Procedure

Step 1 Log in to the server on which the on-premises databases are deployed.

Assume that **/path/to/mongo** is the directory for restoration, and **/path/to/mongo/data** is the directory for storing the backup file.

Step 2 Before the restoration, ensure that the **/path/to/mongo/data** directory is empty.

cd /path/to/mongo/data/

rm -rf *

Step 3 Copy and paste the downloaded backup file package to **/path/to/mongo/data/** and decompress it.

lz4 -d xxx_.tar.gz |tar -xC /path/to/mongo/data/

Step 4 Create the **mongod.conf** configuration file in **/path/to/mongo**.

touch mongod.conf

Step 5 Start the database in single-node mode.

1. Modify the **mongod.conf** file to meet the backup startup configuration requirements.

   The following is a configuration template for backup startup:

   ```
   systemLog:
     destination: file
     path: /path/to/mongo/mongod.log
   ```



```
    logAppend: true
security:
    authorization: enabled
storage:
    dbPath: /path/to/mongo/data
    directoryPerDB: true
    engine: wiredTiger
    wiredTiger:
        collectionConfig: {blockCompressor: snappy}
        engineConfig: {directoryForIndexes: true, journalCompressor: snappy}
        indexConfig: {prefixCompression: true}
net:
    http:
        enabled: false
    port: 27017
    bindIp: xxx.xxx.xxx.xxx,xxx.xxx.xxx.xxx
    unixDomainSocket:
        enabled: false
processManagement:
    fork: true
    pidFilePath: /path/to/mongo/mongod.pid
```

NOTE

bindIp indicates the IP address bound to the database. This field is optional. If it is not specified, your local IP address is bound by default.

2. Run the **mongod.conf** command to start the database.

   */usr/bin/***mongod -f /path/to/mongo/mongod.conf**

   NOTE

   /usr/bin/ is the directory that stores the **mongod** file of the installed MongoDB client.

3. After the database is started, log in to the database using mongo shell to verify the restoration result.

   mongo --host *<DB_HOST>* **-u** *<DB_USER>* **-p** *<PASSWORD>* **--authenticationDatabase admin**

   NOTE

   - **DB_HOST** is the IP address bound to the database.
   - **DB_USER** is the database user. The default value is **rwuser**.
   - **PASSWORD** is the password for the database user, which is the password used for backing up the DB instance.

----End

Starting the Database in Replica Set Mode

By default, the physical backup of the DDS DB instance contains the replica set configuration of the original DB instance. You need to start the database in single-node mode. Otherwise, the database cannot be accessed.

If you want to start the database in replica set mode, perform step **Step 5** and then perform the following steps:

Step 1 Log in to the database using mongo shell.

Step 2 Remove the original replica set configuration.

use local

db.system.replset.remove({})

Step 3 Stop the database process.

use admin

db.shutdownServer()

Step 4 Add the replication configuration in the **mongod.conf** file in the **/path/to/mongo/** directory. For details about the command usage, see **Deploy a Replica Set**.

Step 5 Run the **mongod.conf** command to start the database.

*/usr/bin/***mongod -f /path/to/mongo/mongod.conf**

NOTE

/usr/bin/ is the directory that stores the **mongod** file of the installed MongoDB client.

Step 6 Add the replica set members and initialize the replica set.

NOTE

Use the rs.initiate() command to perform the preceding step. For details, see **rs.initiate()**.

----End

9.6 Restoring Data of Enhanced Edition

Scenarios

DDS 4.4 and DDS 4.4 Enhanced Edition (DDS 4.4 pro for short) are incompatible due to different underlying data storage structures. You can upgrade DDS 4.4 to DDS 4.4 pro in either of the following ways:

- Use DRS to migrate data from DDS 4.4 to DDS 4.4 pro. For details, see .
- You can to upgrade DDS 4.4 to DDS 4.4 pro by rebuilding data on the original DB instance.

Precautions

- After data in a DDS DB instance is migrated to an enhanced binary system, backups before the migration cannot be restored to the original DB instance or a new DB instance.
- After data in a DDS DB instance is migrated to an enhanced binary system, backups cannot be restored to the time point before the migration.
- DDS enhanced binary data backups cannot be restored to on-premises databases.

10 Parameter Template Management

10.1 Overview

DB parameter templates act as a container for engine configuration values that are applied to one or more DB instances. You can customize the parameter settings to manage DB engine configurations.

Parameter Template Type

When creating a DB instance, you can associate a default parameter template or a customized parameter template with the DB instance. After a DB instance is created, you can also change the associated parameter template.

- Default parameter template

  The DB engine parameter values and system service parameter values in the default parameter group are designed for optimizing the database performance.

- Custom parameter template

  If you need a DB instance with customized parameter settings, you can create a parameter template and change the parameter values as required.

  If you change the parameter values of the parameter template associated with several DB instances, the changes will apply to all these DB instances.

Application Scenarios

- If you want to use a customized parameter template, you only need to create a parameter template in advance and select the parameter template when creating a DB instance. For details about how to create a parameter template, see **Creating a Parameter Template**.
- When you have already created a parameter template and want to include most of the custom parameters and values from that template in a new parameter template, you can replicate that parameter template following the instructions provided in section **Replicating a Parameter Template**.

Precautions

- Default parameter templates are unchangeable. You can only view them by clicking their names. If inappropriate settings of customized parameter templates lead to a database startup failure, you can reset the customized parameter template by referring to the settings of the default parameter template.
- After modifying a parameter, you need to view the associated instance status in the instance list. If **Pending restart** is displayed, you need to restart the instance for the modification to take effect.
- Improperly setting parameters in a parameter template may have unintended adverse effects, including degraded performance and system instability. Exercise caution when modifying database parameters and you need to back up data before modifying parameters in a parameter template. Before applying parameter changes to a production DB instance, you should try out these changes on a test DB instance.

10.2 Creating a Parameter Template

Scenarios

A database parameter template acts as a container for engine configuration values that can be applied to one or more DB instances.

Precautions

- DDS does not share parameter template quotas with RDS.
- Each account can create up to 100 DDS parameter templates for the cluster, replica set, and single node instances.

Cluster

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Parameter Templates**.

Step 5 On the **Parameter Templates** page, click **Create Parameter Template**.

Step 6 Select **Cluster** for **DB Instance Type**, specify **DB Engine Version**, **Node Type**, **New Parameter Template**, and **Description** (optional), and then click **OK**.

- **Node Type**: specifies the node type that this parameter template will apply to. For example, to create a parameter template applying to config, select **config**.
- **New Parameter Template**: The template name can be up to 64 characters. It must start with a letter and can contain only letters (case-sensitive), digits, hyphens (-), periods (.), and underscores (_).

- **Description**: It can contain up to 256 characters but cannot contain line breaks or the following special characters >!<"&'=

Step 7 On the **Parameter Templates** page, view and manage parameter templates on the **Clusters** tab.

----End

Replica Set

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Parameter Templates**.

Step 5 On the **Parameter Templates** page, click **Create Parameter Template**.

Step 6 Select **Replica set** for **DB Instance Type**, specify **DB Engine Version**, **Node Type**, **Parameter Template Name**, and **Description** (optional), and then click **OK**.

- **Node Type**: specifies the node type that this parameter template will apply to. For example, to create a parameter template applying to a read replica, select **readonly**.
- **New Parameter Template**: The template name can be up to 64 characters. It must start with a letter and can contain only letters (case-sensitive), digits, hyphens (-), periods (.), and underscores (_).
- **Description**: It can contain up to 256 characters but cannot contain line breaks or the following special characters >!<"&'=

Step 7 On the **Parameter Templates** page, view and manage parameter templates on the **Replica Sets** tab.

----End

Single Node

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Parameter Templates**.

Step 5 On the **Parameter Templates** page, click **Create Parameter Template**.

Step 6 Select **Single node** for **DB Instance Type**, specify **DB Engine Version**, **New Parameter Template**, and **Description** (optional), and then click **OK**.

- **New Parameter Template**: The template name can be up to 64 characters. It must start with a letter and can contain only letters (case-sensitive), digits, hyphens (-), periods (.), and underscores (_).

- **Description**: It can contain up to 256 characters but cannot contain line breaks or the following special characters >!<"&'=

Step 7 On the **Parameter Templates** page, view and manage parameter templates on the **Single Nodes** tab.

----End

10.3 Modifying DDS DB Instance Parameters

Scenarios

You can modify parameters in custom parameter templates as needed to enjoy better performance of DDS.

You can modify parameters in either of the following ways:

- Directly modify the parameters of a specified instance.

  If you modify dynamic parameters on the **Parameters** page of an instance and save the modifications, the modifications take effect immediately regardless of the **Effective upon Reboot** setting. However, if you modify static parameters on the **Parameters** page of an instance and save the modifications, the modifications do not take effect until you manually restart the instance.
- Modify the parameters in a parameter template and apply the template to the instance.

  If you modify parameters in a custom parameter template on the **Parameter Templates** page and save the modifications, the modifications do not take effect until you apply the template to your DB instances. If you modify static parameters in a custom parameter template on the **Parameter Templates** page and save the modifications, the modifications do not take effect until you apply the template to your DB instances and manually restart those DB instances. For details about how to apply a parameter template to instances, see **Applying a Parameter Template**.

Precautions

- You can change parameter values in custom parameter templates but cannot change the default parameter templates provided by the system. You can only click the name of a default parameter template to view its details.
- If a custom parameter template is set incorrectly, the instance associated with the template may fail to start. You can re-configure the custom parameter template according to the configurations of the default parameter template.

CAUTION

Exercise caution when modifying parameter values to prevent exceptions.

Modifying Parameters of an Instance

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Instances**. On the displayed page, click the DB instance whose parameters you wish to modify.

Step 5 In the navigation pane on the left, choose **Parameters**. On the displayed page, modify parameters as required.

Step 6 Modify parameters based on the DB instance type.

- If the DB instance is a cluster instance, select dds mongos, shard, or config on the **Parameters** page and change the value of **net.maxIncomingConnections**, which indicates maximum number of concurrent connections that dds mongos or mongod can be connected.

  Enter **net.maxIncomingConnections** in the search box in the upper right corner of the page and click the search icon to search for this parameter.
- If the DB instance is a replica set instance, select **Replica set nodes** or **Read replicas** on the **Parameters** page and change the value of **net.maxIncomingConnections**, which indicates maximum number of concurrent connections that dds mongos or mongod can be connected.

  Enter **net.maxIncomingConnections** in the search box in the upper right corner of the page and click the search icon to search for this parameter.
- If the DB instance is a single node instance, change the value of **net.maxIncomingConnections**, which indicates maximum number of concurrent connections that dds mongos or mongod can be connected.

  Enter **net.maxIncomingConnections** in the search box in the upper right corner of the page and click the search icon to search for this parameter.

Step 7 Change the maximum number of connections based on the parameter value range and instance specifications. This default value depends on the DB instance specifications. This parameter is displayed as **default** before being set, indicating that the parameter value varies with the memory specifications. For details about the parameters, see **Parameters**.

- To save the changes, click **Save**.
- If you want to cancel the modifications, click **Cancel**.
- If you want to preview the modifications, click **Preview**.

Step 8 After the parameters have been modified, click **Change History** to view parameter modification details. For details, see **Viewing Change History of DB Instance Parameters**.

NOTICE

Check the value in the **Effective upon Restart** column. If it is set to:

- **Yes**: If an instance status on the **Instances** page is **Pending restart**, the instance needs to be restarted to apply changes. If only one node in a replica set, shard, or config is restarted, the changes will not be applied.
- **No**: The changes are applied immediately.

----End

Modifying Parameters in a Custom Parameter Template

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Parameter Templates**.

Step 5 On the **Parameter Templates** page, click **Custom Templates**. Locate the target parameter template and click its name.

Step 6 Modify the required parameters.

For parameter description details, see Parameters.

- If you want to save the modifications, click **Save**.
- If you want to cancel the modifications, click **Cancel**.
- If you want to preview the modifications, click **Preview**.

Step 7 The modifications take effect only after you apply the parameter template to instances. For details, see **Applying a Parameter Template**.

NOTICE

- After the parameters have been modified, click **Change History** to view parameter modification details. For details, see **Viewing Change History of a Custom Parameter Template**.
- The change history page displays only the modifications of the last seven days.
- For details about the parameter template statuses, see **Parameter Template Status**.
- After modifying a parameter, view the associated instance status in the instance list. If **Pending restart** is displayed, restart the instance for the modification to take effect.

----End

10.4 Exporting a Parameter Template

Scenarios

- You can export parameters of a DB instance as a new parameter template for future use. To apply the exported parameter template to new DB instances, see **Applying a Parameter Template**.
- You can export the parameter template details (parameter names, values, and descriptions) of an instance to a CSV file for review and analysis.

Procedure

Step 1 Log in to the management console.

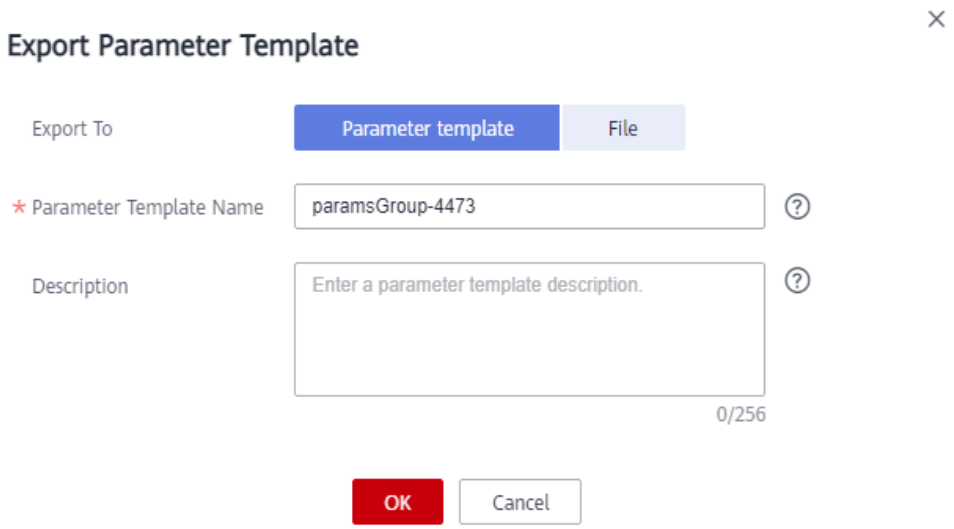
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Instances**. On the displayed page, click the target instance. The **Basic Information** page is displayed.

Step 5 In the navigation pane on the left, choose **Parameters**. On the **Parameters** tab, above the parameter list, click **Export**.

Figure 10-1 Exporting a parameter template



- **Parameter Template**: The parameter list of the instance to will be exported to a parameter template for future use.

  In the displayed dialog box, configure required details and click **OK**.

NOTE

- **New Parameter Template**: The template name can be up to 64 characters. It must start with a letter and can contain only letters (case-sensitive), digits, hyphens (-), periods (.), and underscores (_).
- **Description**: It can contain up to 256 characters but cannot contain line breaks or the following special characters >!<"&'=

After the parameter template is exported, a new template is generated in the list on the **Parameter Templates** page.

- **File**: The parameter template details (parameter names, values, and descriptions) of a DB instance are exported to a CSV file for review and analysis.

  In the displayed dialog box, enter the file name and click **OK**.

  NOTE

  The file name must start with a letter and consist of 4 to 81 characters. It can contain only letters, digits, hyphens (-), and underscores (_).

----End

10.5 Comparing Parameter Templates

Scenarios

This section describes how to compare two parameter templates of the same node type and DB engine version.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

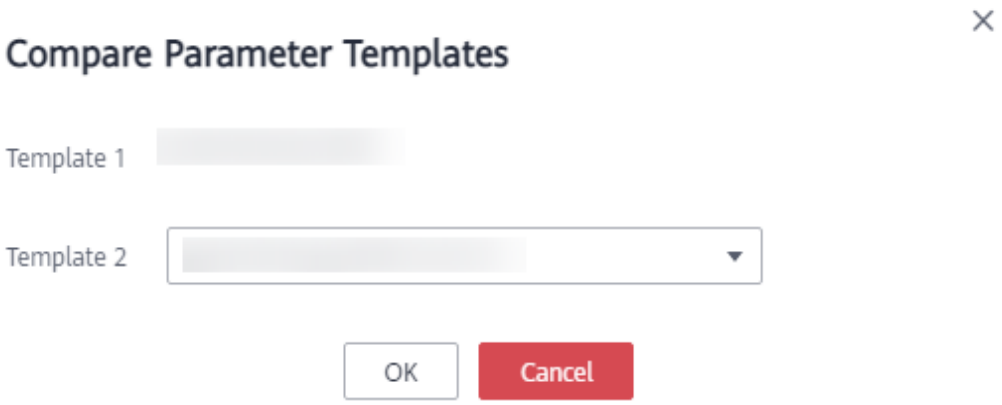
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Parameter Templates**.

Step 5 On the **Parameter Templates** page, locate the parameter template, and click **Compare**.

Step 6 In the displayed dialog box, select a parameter template that uses the same DB engine as the target template and click **OK**.

Figure 10-2 Comparing two parameter templates



- If their settings are different, the parameter names and values of both parameter templates are displayed.
- If their settings are the same, no data is displayed.

----End

10.6 Viewing Parameter Change History

Scenarios

You can view the change history of a parameter template. You can view the change history of both the parameter template used by your DB instance and custom parameter templates.

Precautions

In a newly exported or created parameter template, the change history is blank.

Viewing Change History of DB Instance Parameters

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name. The **Basic Information** page is displayed.

Step 5 In the navigation pane on the left, choose **Parameters**. On the **Change History** tab, view the parameter name, original parameter value, new parameter value, modification status, and modification time.

----End

Viewing Change History of a Custom Parameter Template

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Parameter Templates** page, click **Custom Templates**. Locate the target parameter template and click its name.

Step 5 In the navigation pane on the left, choose **Change History**. Then, view the parameter name, original parameter value, new parameter value, modification status, and modification time.

You can apply the parameter template to DB instances as required by referring to section **Applying a Parameter Template**.

----End

10.7 Replicating a Parameter Template

Scenarios

You can replicate a parameter template you have created. When you have already created a parameter template and want to include most of the custom parameters and values from that template in a new parameter template, you can replicate that parameter template. You can also export the instance's parameters to generate a new template.

Default parameter templates cannot be replicated, but you can create parameter templates based on the default ones.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

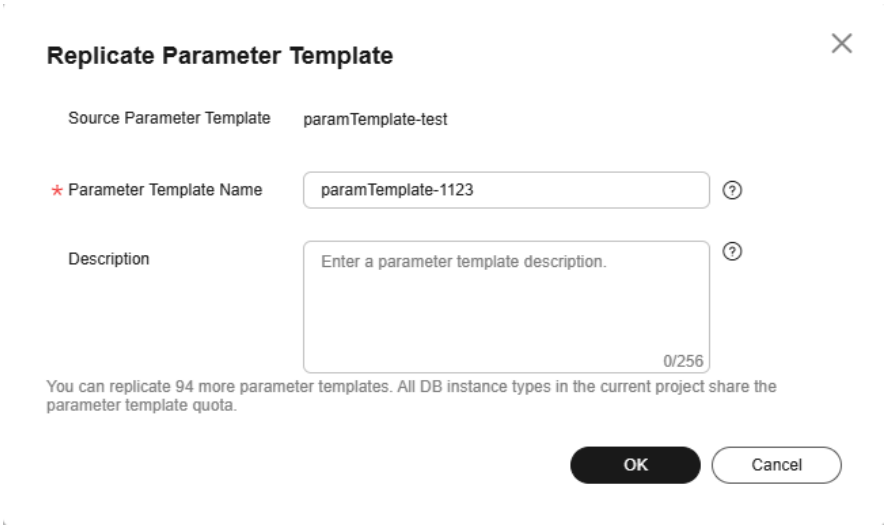
Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Parameter Templates**.

Step 5 On the **Parameter Templates** page, click **Custom Templates**, locate the parameter template, and click **Replicate** in the **Operation** column.

Step 6 In the displayed dialog box, enter the parameter template name and description and click **OK**.

Figure 10-3 Replicating a parameter template



- **Parameter Template Name**: The template name can be up to 64 characters. It can contain only letters, digits, hyphens (-), underscores (_), and periods (.).
- **Description**: The description can contain up to 256 characters but cannot include line breaks or the following special characters >!<"&'=

After the parameter template is replicated, a new template is generated in the list on the **Parameter Templates** page.

----End

10.8 Resetting a Parameter Template

Scenarios

You can reset all parameters in a custom parameter template to their default settings.

Precautions

Resetting a parameter template will restore all parameters in the parameter template to their default values. Exercise caution when performing this operation.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Parameter Templates**.

Step 5 On the **Parameter Templates** page, click **Custom Templates**, locate the parameter template, and choose **More** > **Reset** in the **Operation** column.

Step 6 In the displayed dialog box, click **Yes**.

----End

10.9 Applying a Parameter Template

Scenarios

Modifications to parameters in a custom parameter template take effect for DB instances only after you have applied the template to the DB instances.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Parameter Templates**.

Step 5 On the **Parameter Templates** page, apply a default template or a custom template to the DB instance:

- To apply a default template, click the **Default Templates** tab, locate the required parameter template, and click **Apply** in the **Operation** column.
- To apply a custom template, click **Custom Templates**, locate the parameter template, and in the **Operation** column, choose **More** > **Apply**.

A parameter template can be applied to one or more nodes and instances.

Step 6 In the displayed dialog box, select the node or instance to which the parameter template will be applied and click **OK**.

After the parameter template is successfully applied, you can view the application records by referring to **Viewing Application Records of a Parameter Template**.

----End

10.10 Viewing Application Records of a Parameter Template

Scenarios

You can view the application records of a parameter template.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Parameter Templates**.

Step 5 On the **Parameter Templates** page, select the parameter template for which you want to view application records.

- Click **Default Templates**. Locate the parameter template and click **View Application Record**.
- Click **Custom Templates**. Locate the parameter template and choose **More** > **View Application Record**.

Step 6 You can view the name or ID of the DB instance that the parameter template applies to, as well as the application status, application time, and the causes of any failures that have occurred.

----End

10.11 Modifying the Description of a Parameter Template

Scenarios

The section describes how to modify the description of a parameter template you created so that you can distinguish and identify parameter templates.

Precautions

The description of a default parameter template cannot be modified.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Parameter Templates**.

Step 5 On the **Parameter Templates** page, locate the parameter template, and click in the **Description** column.

Step 6 Enter new description information. The parameter template description can contain up to 256 characters but cannot contain line breaks or the following special characters >!<"&'=

- To submit the change, click ✓. After the modification is successful, you can view the new description in the **Description** column of the parameter template list.
- To cancel the change, click ✗.

----End

10.12 Deleting a Parameter Template

You can delete a custom parameter template that is no longer used.

Precautions

- Default parameter templates and parameter templates applied to instances cannot be deleted.
- Deleted parameter templates cannot be restored. Exercise caution when performing this operation.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Parameter Templates**.

Step 5 On the **Parameter Templates** page, locate the parameter template you want to delete, and choose **More** > **Delete**.

Step 6 In the displayed dialog box, click **Yes**.

----End

11 Database Usage

11.1 Creating a Database Account Using Commands

Scenarios

When you create a DDS instance, the system automatically creates the default account **rwuser**. You can use the default account **rwuser** to create other database accounts based on service requirements. Then, you can use the default account **rwuser** or other created accounts to perform operations on data in the database, such as databases, tables, and indexes.

Precautions

- When creating a database account for a specified instance, you are advised to enable SSL to improve data security.
- If the existing DDS instances are of version 3.2, you cannot create database accounts for them. You can only change the password of the administrator account **rwuser**.
- When creating a database account, configure **passwordDigestor:"server"**. For details, see the **official document**.

Prerequisites

A DDS instance has been connected. For details, see "Connecting to an Instance over a Public Network" and "Connecting to an Instance over a Private Network" in *Document Database Service User Guide*.

Account Description

- When a DDS instance is created, users **root**, **monitor**, and **backup** are automatically created. These accounts belong to the Huawei Cloud DB instance management platform and cannot be operated or used. Attempting to delete, rename, change the passwords, or change privileges for these accounts will result in errors.
- You can change the password of the database administrator **rwuser** and any accounts you create.

- The default user **rwuser** and users created by **rwuser** have limited permissions on system databases **admin** and **config**. They have all required permissions on the databases and tables created under them.
- Generally, a MongoDB user is created in a specified authentication database. When connecting to a database, use **--authenticationDatabase** to specify the corresponding authentication database.
- In a DDS instance, the default authentication database of user **rwuser** is **admin**.
- If you enter incorrect passwords for five consecutive times, the account will be locked for 10s.

Setting Password Strength for Database Accounts

- The administrator password must meet the following password policy:
  - Contains 8 to 32 characters.
  - Must be a combination of uppercase letters, lowercase letters, digits, and special characters: ~!@#%^*-_=+?()$
- The database user created on the client must meet the following password policy:
  - Contains 8 to 32 characters.
  - Must be a combination of uppercase letters, lowercase letters, digits, and special characters: ~@#%-_!*+=^?

When you create a DB instance or set a password, DDS automatically checks your password strength. If the password does not meet the complexity requirements, change the password as prompted.

Creating an Account

Step 1 Run the following command to select the admin database:

use admin

Step 2 Run the following command to create a database account (**user1** as an example):

db.createUser({user: "user1", pwd: "****", passwordDigestor:"server", roles: [{role: "root", db: "admin"}]})**

- ***server*** indicates the password encrypted on the server. It has a fixed value and does not need to be changed.
- **********: indicates the example new password. The password must be 8 to 32 characters in length and contain uppercase letters, lowercase letters, digits, and special characters, such as ~@#%-_!*+=^?
- ***roles*** restricts the rights of the account. If an empty array is specified, the account does not have any permission.

Step 3 Check the result:

The account is successfully created if the following information is displayed:

```
Successfully added user: {
	"user" : "user1",
	"passwordDigestor" : "server",
	"roles" : [
```



```
            {
                "role" : "root",
                "db" : "admin"
            }
        ]
}
```

----End

Changing a Password

Step 1 Run the following command to select the admin database:

use admin

Step 2 Uses user **user1** as an example. Run the following command to change its password:

db.updateUser("user1", {passwordDigestor:"server",pwd:"newPasswd12#"})

- ***server*** indicates the password encrypted on the server. It has a fixed value and does not need to be changed.
- ***newPasswd12#***: indicates the example new password. The password must be 8 to 32 characters in length and contain uppercase letters, lowercase letters, digits, and special characters, such as ~@#%-_!*+=^?
- If the password contains any of the special characters @/%?# and is used in the MongoDB URL, escape the special characters in the URL and replace them with hexadecimal URL codes (ASCII codes).

Step 3 Check the setting result. The password is successfully changed if the following information is displayed:

- Cluster
  ```
  mongos>
  ```
- Replica set
  ```
  replica:PRIMARY>
  ```
- Single node
  ```
  replica:PRIMARY>
  ```

----End

Connecting to an Instance Using the Created Account

After a database account is created, it can be used to connect to the database. The operation details are as follows:

11.2 Creating a Database Using Commands

Scenarios

A database is a collection of tables, indexes, views, stored procedures, and operators. To make it easier to manage DDS DB instances, you can create a database by running commands on the newly-created DB instance. If the database does not exist, create the database and switch to the new database. If the database exists, directly switch to the database.

Prerequisites

A DDS instance has been connected. For details, see "Connecting to an Instance over a Public Network" and "Connecting to an Instance over a Private Network" in *Document Database Service Getting Started.*

Procedure

Step 1 Create a database.

use *dbname*

dbname: indicates the name of the database to be created.

Figure 11-1 Creating databases

```
replica:PRIMARY> use test001
switched to db test001
```

Step 2 After a database is created, insert data into the database so that you can view the database in the database list.

Figure 11-2 Inserting data

```
replica:PRIMARY> db.user.insert({"key1":"value1"})
WriteResult({ "nInserted" : 1 })
replica:PRIMARY> show dbs
admin    0.000GB
local    0.004GB
test     0.000GB
test001  0.000GB
replica:PRIMARY>
```

NOTE

There are three system databases created by default: admin, local, and test. If you directly insert data without creating a database, the data is inserted to the test database by default.

Figure 11-3 Viewing the database

```
replica:PRIMARY> show dbs
admin  0.000GB
local  0.004GB
test   0.000GB
```

Step 3 View data in the database.

Figure 11-4 Viewing data

```
replica:PRIMARY> show collections
user
replica:PRIMARY> db.user.find()
{ "_id" : ObjectId("5da1880d2b4ccf2e1163ad1d"), "key1" : "value1" }
```

----End

11.3 Which Commands are Supported or Restricted by DDS?

The following tables list the commands supported and restricted by DDS.

For more information, see **official MongoDB documentation**.

NOTE

As shown in the following table, "√" indicates that the current version supports the command, and "x" indicates that the current version does not support the command.

Table 11-1 Commands supported and restricted by DDS

Type	Command	3.4	4.0	4.2	Description
Aggregates Commands	aggregate	√	√	√	-
	count	√	√	√	-
	distinct	√	√	√	-
	group	√	√	√	-
	mapReduce	√	√	√	This command can be used only when the **security.javascriptEnabled** parameter in the parameter template associated with the DB instance is set to **true**. For more information, see **How Do I Use MapReduce Commands?**
Geospatial Commands	geoNear	√	√	√	-
	geoSearch	√	√	√	-
Query and Write Operation Commands	find	√	√	√	-
	insert	√	√	√	-
	update	√	√	√	-
	delete	√	√	√	-

Type	Command	3.4	4.0	4.2	Description
	findAndModify	√	√	√	-
	getMore	√	√	√	-
	getLastError	√	√	√	-
	resetError	√	√	√	-
	getPrevError	√	√	√	-
	parallelCollectionScan	√	√	√	-
Query Plan Cache Commands	planCacheListFilters	√	√	√	-
	planCacheSetFilter	√	√	√	-
	planCacheClearFilters	√	√	√	-
	planCacheListQueryShapes	√	√	√	-
	planCacheListPlans	√	√	√	-
	planCacheClear	√	√	√	-
Authentication Commands	logout	√	√	√	-
	authenticate	√	√	√	-
	copydbgetnonce	√	√	√	-
	getnonce	√	√	√	-
	authSchemaUpgrade	x	x	x	System command
User Management Commands	createUser	√	√	√	-
	updateUser	√	√	√	-
	dropUser	√	√	√	-
	dropAllUsersFromDatabase	√	√	√	-
	grantRolesToUser	√	√	√	-
	revokeRolesFromUser	√	√	√	-

Type	Command	3.4	4.0	4.2	Description
	usersInfo	√	√	√	-
Role Management Commands	invalidateUserCache	√	√	√	-
	createRole	√	√	√	-
	updateRole	√	√	√	-
	dropRole	√	√	√	-
	dropAllRolesFromDatabase	√	√	√	-
	grantPrivilegesToRole	√	√	√	-
	revokePrivilegesFromRole	√	√	√	-
	grantRolesToRole	√	√	√	-
	revokeRolesFromRole	√	√	√	-
	rolesInfo	√	√	√	-
Replication Commands	replSetElect	x	x	x	System command
	replSetUpdatePosition	x	x	x	System command
	appendOplogNote	x	x	x	System command
	replSetFreeze	x	x	x	System command
	replSetGetStatus	√	√	√	-
	replSetInitiate	x	x	x	System command
	replSetMaintenance	x	x	x	System command
	replSetReconfig	x	x	x	System command
	replSetStepDown	x	x	x	System command
	replSetSyncFrom	x	x	x	System command

Type	Command	3.4	4.0	4.2	Description
	replSetRequestVotes	x	x	x	System command
	replSetDeclareElectionWinner	x	x	x	System command
	resync	x	x	x	System command
	applyOps	x	x	x	System command
	isMaster	√	√	√	-
	replSetGetConfig	x	x	x	System command
Sharding Commands	flushRouterConfig	√	√	√	High-risk commands
	addShard	x	x	x	Unauthorized operation
	addShardToZone	√	√	√	-
	balancerStart	√	√	√	-
	balancerStatus	√	√	√	-
	balancerStop	√	√	√	-
	removeShardFromZone	√	√	√	-
	updateZoneKeyRange	√	√	√	-
	cleanupOrphaned	x	x	x	High-risk commands
	checkShardingIndex	x	x	x	System command
	enableSharding	√	√	√	-
	listShards	x	x	x	System command
	removeShard	x	x	x	High-risk commands
	getShardMap	x	x	x	System command
	getShardVersion	√	√	√	-

Type	Command	3.4	4.0	4.2	Description
	mergeChunks	√	√	√	-
	setShardVersion	x	x	x	System command
	shardCollection	√	√	√	-
	shardingState	x	x	x	System command
	unsetSharding	x	x	x	System command
	split	√	√	√	-
	splitChunk	√	√	√	-
	splitVector	√	√	√	-
	moveChunk	√	√	√	-
	movePrimary	√	√	√	-
	isdbgrid	√	√	√	-
Administration Commands	setFeatureCompatibilityVersion	√	√	√	-
	renameCollection	√	√	√	-
	dropDatabase	√	√	√	-
	listCollections	√	√	√	-
	drop	√	√	√	-
	create	√	√	√	-
	clone	x	x	x	System command
	cloneCollection	√	√	√	-
	cloneCollectionAsCapped	√	√	√	-
	convertToCapped	√	√	√	-
	filemd5	√	√	√	-
	createIndexes	√	√	√	-
	listIndexes	√	√	√	-
	dropIndexes	√	√	√	-

Type	Command	3.4	4.0	4.2	Description
	fsync	√	√	√	-
	clean	x	x	x	System command
	connPoolSync	x	x	x	System command
	connectionStatus	√	√	√	-
	compact	x	x	x	High-risk commands
	collMod	√	√	√	-
	reIndex	√	√	√	-
	setParameter	x	x	x	System configuration command
	getParameter	√	√	√	-
	repairDatabase	x	x	x	High-risk commands
	repairCursor	x	x	x	System command
	touch	√	√	√	-
	shutdown	x	x	x	High-risk commands
	logRotate	x	x	x	High-risk commands
	killOp	√	√	√	-
	releaseFreeMemory	√	√	√	-
Diagnostic Commands	availableQueryOptions	√	√	√	-
	buildInfo	√	√	√	-
	collStats	√	√	√	-
	connPoolStats	x	x	x	System command
	cursorInfo	x	x	x	System command
	dataSize	√	√	√	-

Type	Command	3.4	4.0	4.2	Description
	dbHash	x	x	x	System command
	dbStats	√	√	√	-
	diagLogging	x	x	x	System command
	driverOIDTest	x	x	x	System command
	explain	√	√	√	-
	features	√	√	√	-
	getCmdLineOpts	x	x	x	System command
	getLog	x	x	x	System command
	hostInfo	x	x	x	System command
	isSelf	x	x	x	System command
	listCommands	√	√	√	-
	listDatabases	√	√	√	-
	netstat	x	x	x	System command
	ping	√	√	√	-
	profile	√	√	√	-
	serverStatus	√	√	√	-
	shardConnPoolStats	x	x	x	System command
	top	√	√	√	-
	validate	x	x	x	System configuration command
	whatsmyuri	√	√	√	-
Internal Commands	handshake	x	x	x	System command
	_recvChunkAbort	x	x	x	System command

Type	Command	3.4	4.0	4.2	Description
	_recvChunkCommit	x	x	x	System command
	_recvChunkStart	x	x	x	System command
	_recvChunkStatus	x	x	x	System command
	_replSetFresh	x	x	x	System command
	mapreduce.shardedfinish	x	x	x	System command
	_transferMods	x	x	x	System command
	replSetHeartbeat	x	x	x	System command
	replSetGetRBID	x	x	x	System command
	_migrateClone	x	x	x	System command
	replSetElect	x	x	x	System command
	writeBacksQueued	x	x	x	System command
	writebacklisten	x	x	x	System command
System Events Auditing Commands	logApplicationMessage	x	x	x	System command

12 Data Security

12.1 Enabling or Disabling SSL

Secure Socket Layer (SSL) is an encryption-based Internet security protocol for establishing an encrypted link between a server and a client. It provides privacy, authentication, and integrity to Internet communications.

- Authenticates users and servers, ensuring that data is sent to the correct clients and servers.
- Encrypts data to prevent it from being intercepted during transfer.
- Ensures data integrity during transmission.

After SSL is enabled, you can establish an encrypted connection between your client and the instance you want to access to improve data security.

Precautions

- Enabling or disabling SSL will cause instances to restart. Exercise caution when performing this operation.

  NOTE

  When you enable or disable SSL, DDS will restart once. During the restart, each node will be intermittently disconnected for about 30 seconds. You are advised to enable or disable SSL during off-peak hours and ensure that your applications support automatic reconnection.

- If SSL is enabled, you can connect to a database using SSL, which is more secure.

  Currently, insecure encryption algorithms are disabled. The following table lists the supported TLS versions and cipher suites.

Version	TLS Version	Cipher Suites
4.0	TLS 1.2	DHE-RSA-AES256-GCM-SHA384 DHE-RSA-AES128-GCM-SHA256
4.2	TLS 1.2	DHE-RSA-AES256-GCM-SHA384 DHE-RSA-AES128-GCM-SHA256

Version	TLS Version	Cipher Suites
4.4	TLS 1.2	DHE-RSA-AES256-GCM-SHA384 DHE-RSA-AES128-GCM-SHA256
5.0	TLS 1.2	DHE-RSA-AES256-GCM-SHA384 DHE-RSA-AES128-GCM-SHA256

The server where the client is located must support the corresponding TLS version and encryption algorithm suite. Otherwise, the connection fails.

- If SSL is disabled, you can connect to a database using an unencrypted connection.

Enabling SSL

Step 1 Log in to the management console.

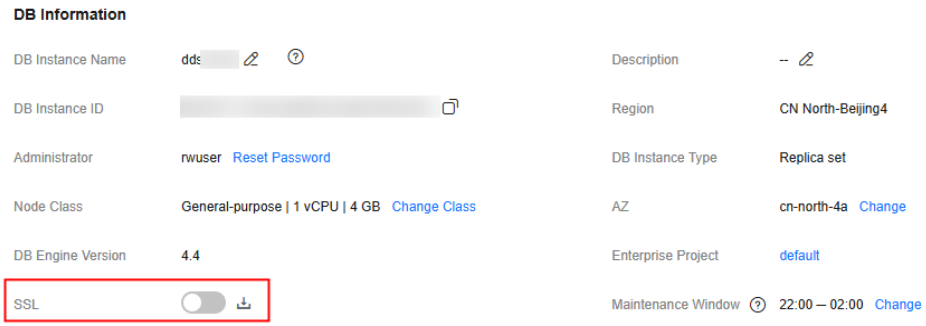
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the target DB instance.

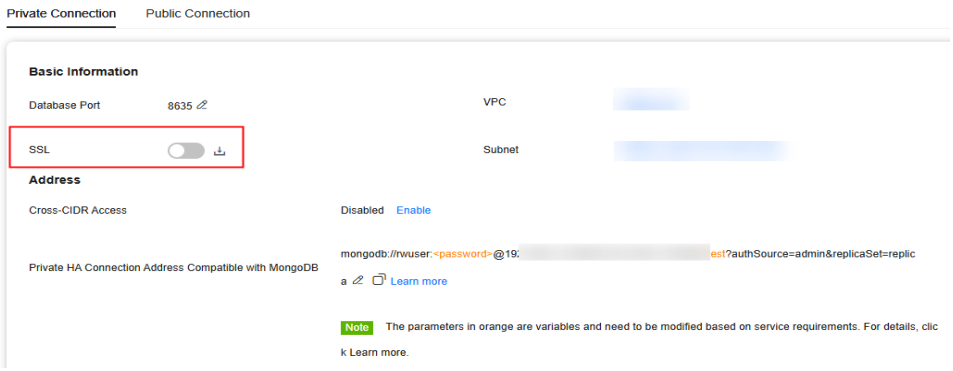
Step 5 In the **DB Information** area on the **Basic Information** page, click next to the **SSL** field.

Figure 12-1 Enabling SSL



Alternatively, in the navigation pane on the left, choose **Connections**. In the **Private Connection** area, click next to the **SSL** field.

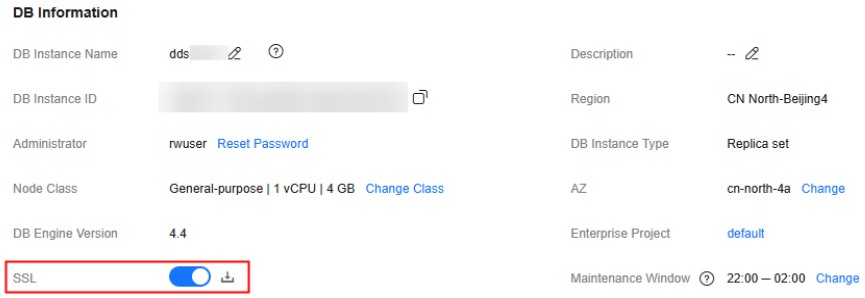
Figure 12-2 Enabling SSL



Step 6 In the displayed dialog box, click **Yes**.

Step 7 In the **DB Information** area, view the modification result.

Figure 12-3 SSL enabled



Step 8 After SSL is enabled, click next to **SSL** to download an SSL certificate.

For details about how to connect to an instance using SSL, refer to the following content:

- **Connecting to a Cluster Instance Using SSL**
- **Connecting to a Replica Set Instance Using SSL**
- **Connecting to a Single Node Instance Using SSL**

----End

Disabling SSL

Step 1 Log in to the management console.

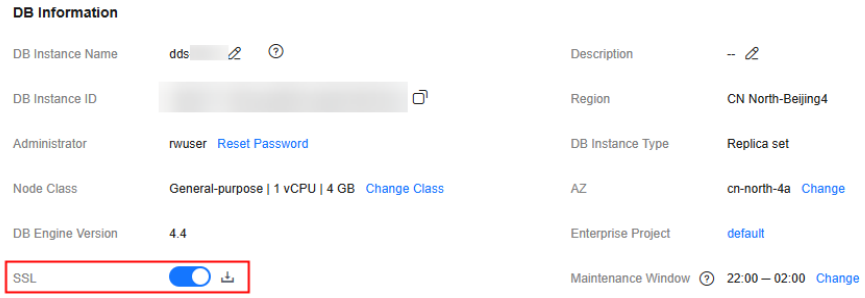
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the target DB instance.

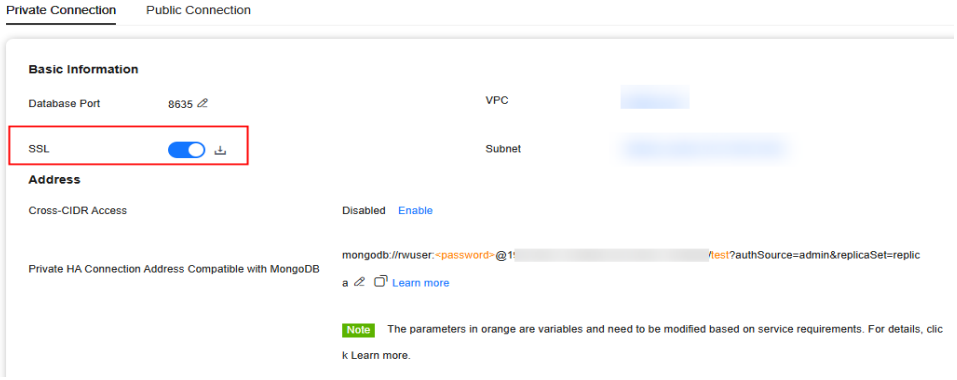
Step 5 In the **DB Information** area on the **Basic Information** page, click next to the **SSL** field.

Figure 12-4 Disabling SSL



Alternatively, in the navigation pane on the left, choose **Connections**. In the **Basic Information** area, click  next to the **SSL** field.

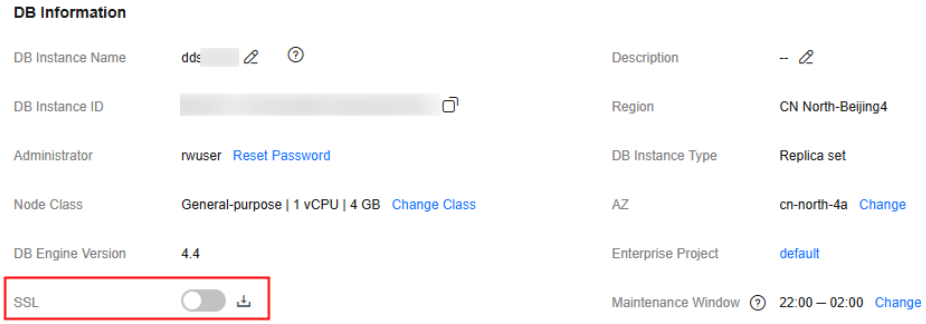
Figure 12-5 Disabling SSL



Step 6 In the displayed dialog box, click **Yes**.

Step 7 In the **DB Information** area, view the modification result.

Figure 12-6 SSL disabled



Step 8 Connect to an instance using an unencrypted connection.

For details, refer to the following content:

- **Connecting to a Cluster Instance Using an Unencrypted Connection**
- **Connecting to a Replica Set Instance Using an Unencrypted Connection**
- **Connecting to a Single Node Instance Using an Unencrypted Connection**

----End

12.2 Resetting the Administrator Password

Scenarios

For security reasons, you are advised to periodically change administrator passwords.

If you do not set an administrator password for the DB instance that you are creating, you need to reset the password before connecting to the DB instance.

Precautions

- You cannot reset the administrator password for an instance is in any of the following statuses:
  - Frozen
  - Creating
  - Restarting
  - Adding node
  - Switching SSL
  - Changing port
  - Changing instance class
  - Deleting node
  - Upgrading minor version
  - Switchover in progress
  - Changing AZ
  - Adding read replicas
- If you enable operation protection to improve the security of your account and cloud products, two-factor authentication is required for sensitive operations. For details about how to enable operation protection, see Operation Protection in *Identity and Access Management User Guide*.

CAUTION

Changing the password may interrupt services.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the target DB instance and choose **More** > **Reset Password** in the **Operation** column.

Alternatively, on the **Instances** page, click the instance. In the **DB Information** area on the **Basic Information** page, click **Reset Password** in the **Administrator** field.

Step 5 Enter and confirm the new administrator password and click **OK**.

- Resetting the password does not disconnect the authenticated connection. However, you will need to enter the new password when logging in to the database.
- The password must be 8 to 32 characters in length and contain uppercase letters, lowercase letters, digits, and any of the following special characters ~!@#%^*-_=+?()$

Step 6 If you have enabled operation protection, click **Start Verification** in the displayed dialog box. On the displayed page, click **Send Code**, enter the verification code, and click **Verify** to close the page.

----End

12.3 Changing a Security Group

This section describes how to change a security group for cluster and replica set instances

Precautions

If any of the following operations is in progress, do not change the security group:

- Adding nodes
- Migrating data

Changing a Security Group

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

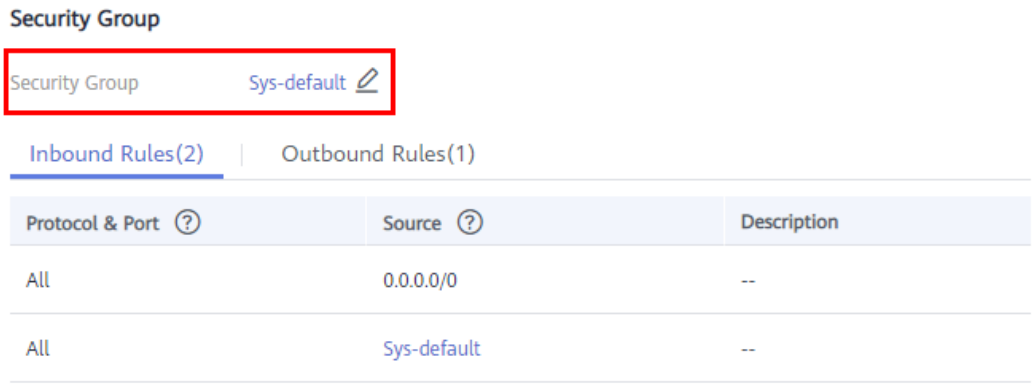
Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the target DB instance.

Step 5 In the navigation pane on the left, choose **Connections**.

Step 6 In the **Security Group** area, click  to select the security group to which the DB instance belongs.

Figure 12-7 Changing a security group



- To submit the change, click ✓. This process takes about 1 to 3 minutes.
- To cancel the change, click ✗.

Step 7 View the modification result.

----End

13 Monitoring and Alarm Reporting

13.1 DDS Metrics

Description

This section describes the DDS metrics that can be monitored by Cloud Eye as well as their namespaces and dimensions. You can use **APIs** provided by Cloud Eye to query the DDS metrics and alarms.

NOTE

Cloud Eye can monitor dimensions (or objects) nested to a maximum depth of four levels (levels 0 to 3). 3 is the deepest level. For example, if the monitored dimension of a metric is **mongodb_instance_id,mongodb_node_id**, **mongodb_instance_id** indicates level 0 and **mongodb_node_id** indicates level 1.

Namespace

SYS.DDS

Monitoring Metrics

Table 13-1 Recommended DDS metrics

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo007_connections_usage	Percentage of Active Node Connections	Percentage of the number of connections that attempt to connect to the instance node to the total number of available connections	0~100	%	N/A	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute 5 seconds
mongo032_mem_usage	Memory Usage	Memory usage of the monitored object	0~100	%	N/A	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute 5 seconds

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo031_cpu_usage	CPU Usage	CPU usage of the monitored object	0~100	%	N/A	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute 5 seconds
mongo035_disk_usage	Storage Space Usage	Storage usage of the monitored object	0~100	%	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute

Table 13-2 DDS metrics

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo001_command_ps	COMMAND Statements per Second	Number of COMMAND statements executed per second on the current node	≥ 0	Count/s	N/A	mongodb_instance_id,mongodb_node_id	• DDS DB instance • dds mongos node • Read replica of a DDS replica set instance • Primary node • Secondary node • Hidden node	1 minute 5 seconds
mongo002_delete_ps	DELETE Statements per Second	Number of DELETE statements executed per second on the current node	≥ 0	Count/s	N/A	mongodb_instance_id,mongodb_node_id	• DDS DB instance • dds mongos node • Primary node • Secondary node	1 minute 5 seconds
mongo003_insert_ps	INSERT Statements per Second	Number of INSERT statements executed per second on the current node	≥ 0	Count/s	N/A	mongodb_instance_id,mongodb_node_id	• DDS DB instance • dds mongos node • Primary node • Secondary node	1 minute 5 seconds

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo004_query_ps	QUERY Statements per Second	Number of QUERY statements executed per second on the current node	≥ 0	Count/s	N/A	mongodb_instance_id,mongodb_node_id	• DDS DB instance • dds mongos node • Primary node • Secondary node	1 minute 5 seconds
mongo005_update_ps	UPDATE Statements per Second	Number of UPDATE statements executed per second on the current node	≥ 0	Count/s	N/A	mongodb_instance_id,mongodb_node_id	• DDS DB instance • dds mongos node • Primary node • Secondary node	1 minute 5 seconds
mongo006_getmore_ps	GETMORE Statements per Second	Number of GETMORE statements executed per second on the current node	≥ 0	Count/s	N/A	mongodb_instance_id,mongodb_node_id	• DDS DB instance • dds mongos node • Primary node • Secondary node	1 minute 5 seconds
mongo007_chunk_num1	Chunks of Shard 1	Number of chunks in shard 1	0~64	Count	N/A	mongodb_instance_id	DDS cluster instance	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo007_chunk_num2	Chunks of Shard 2	Number of chunks in shard 2	0~64	Count	N/A	mongodb_instance_id	DDS cluster instance	1 minute
mongo007_chunk_num3	Chunks of Shard 3	Number of chunks in shard 3	0~64	Count	N/A	mongodb_instance_id	DDS cluster instance	1 minute
mongo007_chunk_num4	Chunks of Shard 4	Number of chunks in shard 4	0~64	Count	N/A	mongodb_instance_id	DDS cluster instance	1 minute
mongo007_chunk_num5	Chunks of Shard 5	Number of chunks in shard 5	0~64	Count	N/A	mongodb_instance_id	DDS cluster instance	1 minute
mongo007_chunk_num6	Chunks of Shard 6	Number of chunks in shard 6	0~64	Count	N/A	mongodb_instance_id	DDS cluster instance	1 minute
mongo007_chunk_num7	Chunks of Shard 7	Number of chunks in shard 7	0~64	Count	N/A	mongodb_instance_id	DDS cluster instance	1 minute
mongo007_chunk_num8	Chunks of Shard 8	Number of chunks in shard 8	0~64	Count	N/A	mongodb_instance_id	DDS cluster instance	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo007_chunk_num9	Chunks of Shard 9	Number of chunks in shard 9	0~64	Count	N/A	mongodb_instance_id	DDS cluster instance	1 minute
mongo007_chunk_num10	Chunks of Shard 10	Number of chunks in shard 10	0~64	Count	N/A	mongodb_instance_id	DDS cluster instance	1 minute
mongo007_chunk_num11	Chunks of Shard 11	Number of chunks in shard 11	0~64	Count	N/A	mongodb_instance_id	DDS cluster instance	1 minute
mongo007_chunk_num12	Chunks of Shard 12	Number of chunks in shard 12	0~64	Count	N/A	mongodb_instance_id	DDS cluster instance	1 minute
mongo008_connections	Active Instance Connections	Total number of connections attempting to connect to a DDS DB instance	0~200	Count	N/A	mongodb_instance_id	DDS DB instance	1 minute
mongo009_migFail_num	Chunk Migration Failures in Last 24 hrs	Number of chunk migration failures in the last 24 hours	≥ 0	Count	N/A	mongodb_instance_id	DDS cluster instance	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo007_connections	Active Node Connections	Total number of connections attempting to connect to a DDS DB instance node	0~200	Count	N/A	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute 5 seconds
mongo007_connections_usage	Percentage of Active Node Connections	Percentage of the number of connections that attempt to connect to the instance node to the total number of available connections	0~100	%	N/A	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute 5 seconds
mongo008_mem_resident	Resident Memory	Size of resident memory	≥ 0	MB	N/A	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo009_mem_virtual	Virtual Memory	Size of virtual memory	≥ 0	MB	N/A	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute
mongo010_regular_asserts_ps	Regular Asserts per Second	Number of regular asserts per second	≥ 0	Count/s	N/A	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute
mongo011_warning_asserts_ps	Warning Asserts per Second	Number of warning asserts per second	≥ 0	Count/s	N/A	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute
mongo012_msg_asserts_ps	Message Asserts per Second	Number of message asserts per second	≥ 0	Count/s	N/A	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute
mongo013_user_asserts_ps	User Asserts per Second	Number of user asserts per second	≥ 0	Count/s	N/A	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo014_queues_total	Operations Queued Waiting for a Lock	Number of operations queued waiting for a lock	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo015_queues_readers	Operations Queued Waiting for a Read Lock	Number of operations queued waiting for a read lock	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo016_queues_writers	Operations Queued Waiting for a Write Lock	Number of operations queued waiting for a write lock	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo017_page_faults	Page Faults	Number of page faults on the current node	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo018_porfling_num	Slow Queries	Total number of slow queries from the last 5 minutes to the current time point on the monitored node.	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo019_cursors_open	Maintained Cursors	Number of maintained cursors on the current node	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo020_cursors_timeOut	Timeout Cursors	Number of timed out cursors on the current node	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo021_wt_cache_usage	Bytes in WiredTiger Cache	Size of data in the WiredTiger cache in MB	≥ 0	MB	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo022_wt_cache_dirty	Tracked Dirty Bytes in WiredTiger Cache	Size of tracked dirty data in the WiredTiger cache in MB	≥ 0	MB	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo023_wInto_wtCache	Bytes Written Into Cache per Second	Bytes written into WiredTiger cache per second	≥ 0	byte/s	1024(IEC)	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo024_wFrom_wtCache	Bytes Written From Cache per Second	Bytes written into disks from WiredTiger cache per second	≥ 0	byte/s	1024(IEC)	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo025_repl_oplog_win	Oplog Window	Available time in the monitored primary node's oplog	≥ 0	h	N/A	mongodb_node_id	Primary node	1 minute
mongo025_repl_headroom	Replication Headroom	Time difference between the primary's oplog window and the replication lag of the secondary	≥ 0	s	N/A	mongodb_node_id	Secondary node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo026_repl_lag	Replication Lag	A delay between an operation on the primary and the application of that operation from the oplog to the secondary	≥ 0	s	N/A	mongodb_node_id	Secondary node	1 minute
mongo027_repl_command_ps	Replicated COMMAND Statements per Second	Number of replicated COMMAND statements executed on the secondary node per second	≥ 0	Count/s	N/A	mongodb_node_id	Secondary node	1 minute
mongo028_repl_update_ps	Replicated UPDATE Statements per Second	Number of replicated UPDATE statements executed on the secondary node per second	≥ 0	Count/s	N/A	mongodb_node_id	Secondary node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo029_repl_delete_ps	Replicated DELETE Statements per Second	Number of replicated DELETE statements executed on the secondary node per second	≥ 0	Count/s	N/A	mongodb_node_id	Secondary node	1 minute
mongo030_repl_insert_ps	Replicated INSERT Statements per Second	Number of replicated INSERT statements executed on the secondary node per second	≥ 0	Count/s	N/A	mongodb_node_id	Secondary node	1 minute
mongo031_cpu_usage	CPU Usage	CPU usage of the monitored object	0~100	%	N/A	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute 5 seconds
mongo032_mem_usage	Memory Usage	Memory usage of the monitored object	0~100	%	N/A	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute 5 seconds

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo033_bytes_out	Network Output Throughput	Outgoing traffic in bytes per second	≥ 0	byte/s	1024(IEC)	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute 5 seconds
mongo034_bytes_in	Network Input Throughput	Incoming traffic in bytes per second	≥ 0	byte/s	1024(IEC)	mongodb_node_id	• dds mongos node • Primary node • Secondary node	1 minute 5 seconds
mongo035_disk_usage	Storage Space Usage	Storage usage of the monitored object	0~100	%	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo036_iops	IOPS	Average number of I/O requests processed by the system in a specified period	≥ 0	Count/s	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo037_read_throughput	Disk Read Throughput	Average number of read bytes per second for disks	≥ 0	byte/s	1024(IEC)	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo038_write_throughput	Disk Write Throughput	Average number of write bytes per second for disks	≥ 0	byte/s	1024(IEC)	mongodb_node_id	• Primary node • Secondary node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo039_avg_disk_sec_per_read	Average Time per Disk Read	Average time required for each disk read in a specified period	≥ 0	s	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo040_avg_disk_sec_per_write	Average Time per Disk Write	Average time required for each disk write in a specified period	≥ 0	s	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo042_disk_total_size	Total Storage Space	Total disk size of the monitored object	0~1000	GB	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo043_disk_used_size	Used Storage Space	Used storage space of the monitored object	0~1000	GB	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo044_swap_usage	SWAP Usage	SWAP usage.	0~100	%	N/A	mongodb_node_id	• dds mongos node • Secondary node	1 minute
mongo050_top_total_time	Total Time Spent on Collections	Mongotop-total time: total time spent on collection operations.	≥ 0	ms	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo051_top_read_time	Total Time Spent on Collections	Mongotop-read time: total time spent reading collections.	≥ 0	ms	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo052_top_write_time	Total Time Spent on Collections	Mongotop-write time: total time spent writing collections.	≥ 0	ms	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo053_wt_flushes_status	Number of Times that Checkpoints Are Triggered	Number of times that the checkpoint is triggered during a polling interval of WiredTiger	≥ 0	times	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo054_wt_cache_used_percent	Percentage of the Cache Used by WiredTiger	Percentage of the cache used by WiredTiger	0~100	%	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo055_wt_cache_dirty_percent	Percentage of Dirty Data in the WiredTiger Cache	Percentage of dirty data in the WiredTiger cache	0~100	%	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo070_rocks_active_memtable	Memtable Data Size	Size of data in the active memtable	0~100	Byte	1024(IEC)	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo071_rocks_ologcf_active_memtable	Memtable Data Size on Ologcf	Size of data in the active memtable on ologcf	0~100	Byte	1024(IEC)	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo072_rocks_all_memtable	Total Data Size of Memtable and Immutable-memtable	Total data size of memtable and immutable-memtable	0~100	Byte	1024(IEC)	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo073_rocks_ologcf_all_memtable	Total Data Size of Memtable and Immutable-memtable on Ologcf	Total data size of memtable and immutable-memtable on ologcf	0~100	Byte	1024(IEC)	mongodb_node_id	• Primary node • Secondary node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo074_rocks_snapshots	Unreleased Snapshots	Number of unreleased snapshots	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo075_rocks_oplogcf_snapshots	Unreleased Snapshots on Oplogcf	Number of unreleased snapshots on oplogcf	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo076_rocks_live_versions	Active Versions	Number of active versions	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo077_rocks_oplogcf_live_versions	Active Versions on Oplogcf	Number of active versions on oplogcf	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo078_rocks_block_cache	Data Size in Blockcache	Size of data in blockcache	0~100	Byte	1024(IEC)	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo079_rocks_background_errors	Accumulated Background Errors	Accumulated number of background errors	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo080_rocks_oplogcf_background_errors	Accumulated Background Errors on Oplogcf	Number of accumulated background errors on oplogcf	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo081_rocks_conflict_bytes_usage	Buffer Usage for Processing Transaction Write Conflicts	Usage of the buffer for processing transaction write conflicts	0~100	%	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo082_rocks_uncommitted_keys	Uncommitted Keys	Number of uncommitted keys	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo083_rocks_committed_keys	Committed Keys	Number of committed keys	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo084_rocks_alive_txn	Length of Active Transaction Linked Lists	Length of active transaction linked lists	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo085_rocks_read_queue	Length of Read Queues	Length of read queues	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo086_rocks_commit_queue	Length of Committed Queues	Length of committed queues	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo087_rocks_ct_write_out	Used Concurrent Write Transactions	Number of used concurrent write transactions	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo088_rocks_ct_write_available	Available Concurrent Write Transactions	Number of available concurrent write transactions	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo089_rocks_ct_read_out	Used Concurrent Read Transactions	Number of used concurrent read transactions	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute
mongo090_rocks_ct_read_available	Available Concurrent Read Transactions	Number of available concurrent read transactions	≥ 0	Count	N/A	mongodb_node_id	• Primary node • Secondary node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo091_active_session_count	Active Sessions	Number of active sessions cached in the memory of the Mongo instance since the last refresh	≥ 0	Count	N/A	mongodb_node_id	• Read replica of a DDS replica set instance • Primary node • Secondary node • Hidden nodes of a DDS instance	1 minute
mongo092_rx_errors	Error Rate of Received Packets	Ratio of the number of error packets to the total number of received packets during the monitoring period	0~100	%	N/A	mongodb_node_id	• Primary node • Secondary node • Hidden node	1 minute 5 seconds

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo093_rx_dropped	Loss Rate of Received Packets	Ratio of the number of lost packets to the total number of received packets during the monitoring period	0~100	%	N/A	mongodb_node_id	• Primary node • Secondary node • Hidden node	1 minute 5 seconds
mongo094_tx_errors	Error Rate of Sent Packets	Ratio of the number of error packets to the total number of sent packets during the monitoring period	0~100	%	N/A	mongodb_node_id	• Primary node • Secondary node • Hidden node	1 minute 5 seconds

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo095_tx_dropped	Loss Rate of Sent Packets	Ratio of the number of lost packets to the total number of sent packets during the monitoring period	0~100	%	N/A	mongodb_node_id	• Primary node • Secondary node • Hidden node	1 minute 5 seconds
mongo096_retrans_segs	Retransmitted Packets	The number of retransmitted packets during the monitoring period	≥ 0	Counts	N/A	mongodb_node_id	• Primary node • Secondary node • Hidden node	1 minute 5 seconds
mongo097_retrans_rate	Retransmission Ratio	Ratio of retransmitted packets during the monitoring period	0~100	%	N/A	mongodb_node_id	• Primary node • Secondary node • Hidden node	1 minute 5 seconds

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo098_out_rsts_nums	Sent RST Packets	The number of sent RST packets during the monitoring period	≥ 0	Counts	N/A	mongodb_node_id	• Primary node • Secondary node • Hidden node	1 minute 5 seconds
mongo099_read_time_average	Average Read Latency	Average read command execution latency of a single node	≥ 0	ms	N/A	mongodb_node_id	• dds shard node • dds config node • Primary node • Secondary node • Hidden node	1 minute
mongo100_read_time_p99	P99 Read Latency	P99 read command execution latency of a single node	≥ 0	ms	N/A	mongodb_node_id	• dds shard node • dds config node • Primary node • Secondary node • Hidden node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo101_read_time_p999	P999 Read Latency	P999 read command execution latency of a single node	≥ 0	ms	N/A	mongodb_node_id	• dds shard node • dds config node • Primary node • Secondary node • Hidden node	1 minute
mongo102_write_time_average	Average Write Latency	Average write command execution latency of a single node	≥ 0	ms	N/A	mongodb_node_id	• dds shard node • dds config node • Primary node • Secondary node • Hidden node	1 minute
mongo103_write_time_p99	P99 Write Latency	P99 write command execution latency of a single node	≥ 0	ms	N/A	mongodb_node_id	• dds shard node • dds config node • Primary node • Secondary node • Hidden node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo104_write_time_p999	P999 Write Latency	P999 write command execution latency of a single node	≥ 0	ms	N/A	mongodb_node_id	• dds shard node • dds config node • Primary node • Secondary node • Hidden node	1 minute
mongo105_command_time_average	Average Command Latency	Average command execution latency of a single node	≥ 0	ms	N/A	mongodb_node_id	• dds shard node • dds config node • Primary node • Secondary node • Hidden node	1 minute
mongo106_command_time_p99	P99 Command Latency	P99 command execution latency of a single node	≥ 0	ms	N/A	mongodb_node_id	• dds shard node • dds config node • Primary node • Secondary node • Hidden node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo107_command_time_p999	P999 Command Latency	P999 command execution latency of a single node	≥ 0	ms	N/A	mongodb_node_id	• dds shard node • dds config node • Primary node • Secondary node • Hidden node	1 minute
mongo108_txn_time_average	Average Transaction Latency	Average transaction execution latency of a single node	≥ 0	ms	N/A	mongodb_node_id	• dds shard node • dds config node • Primary node • Secondary node • Hidden node	1 minute
mongo109_txn_time_p99	P99 Transaction Latency	P99 transaction execution latency of a single node	≥ 0	ms	N/A	mongodb_node_id	• dds shard node • dds config node • Primary node • Secondary node • Hidden node	1 minute

Metric ID	Metrics Name	Description	Value Range	Unit	Conversion Rule	Dimension	Monitored Object	Monitoring Interval (Raw Data)
mongo110_txn_time_p999	P999 Transaction Latency	P999 transaction execution latency of a single node	≥ 0	ms	N/A	mongodb_node_id	• dds shard node • dds config node • Primary node • Secondary node • Hidden node	1 minute

NOTE

Metrics whose IDs contain rocks are used to monitor instances or instance nodes of version 4.2.

If an object is in a hierarchical system, specify the monitored dimension in hierarchical form when you use an API to query the metrics of this object.

For example, if you want to query the disk usage (**mongo035_disk_usage**) of a DDS instance node, the dimension information of the metric is **mongodb_instance_id,mongodb_node_id**, indicating that **mongodb_instance_id** is level 0 and **mongodb_node_id** is level 1.

```
dim.0=mongodb_instance_id,74d7c42c0ef94136b2a9aed0c6871b51in02&dim.1=mongodb_node_id,260a129ba25c4bd0a4a5b1f54bb698a5no02
```

Dimensions

Key	Value
mongodb_instance_id	DDS DB instance ID Supports cluster instances of Community Edition, replica set instances, and single node instances.
mongodb_node_id	DDS node ID

NOTE

mongodb_instance_id is used to specify dimension fields when the Cloud Eye API is invoked. Replica sets and single node instance types do not have instance-level metrics.

13.2 Configuring Monitoring by Seconds

The default monitoring interval is 1 minute. To improve the instantaneous accuracy of monitoring metrics, you can set the monitoring interval to 5 seconds.

Precautions

- Only some monitoring metrics support monitoring by seconds. For details, see **Monitoring Metrics**.

Enabling Monitoring by Seconds

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the target instance name.

Step 5 In the navigation pane on the left, choose **Advanced O&M**.

Step 6 On the displayed page, click the **Real-Time Monitoring** tab and click next to **Monitoring by Seconds**.

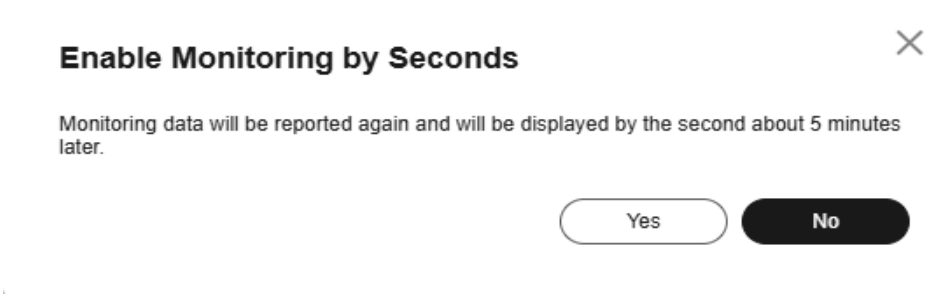
NOTICE

Instances with fewer than four vCPUs do not support monitoring by seconds.

Step 7 In the displayed dialog box, select a collection period and click **Yes**.

Monitoring by Seconds will be automatically disabled for instances with fewer than 4 vCPUs. After you enable this function, monitoring data will be reported again and will be displayed by seconds about five minutes later.

Figure 13-1 Enable Monitoring by Seconds



----End

Disabling Monitoring by Seconds

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the target instance name.

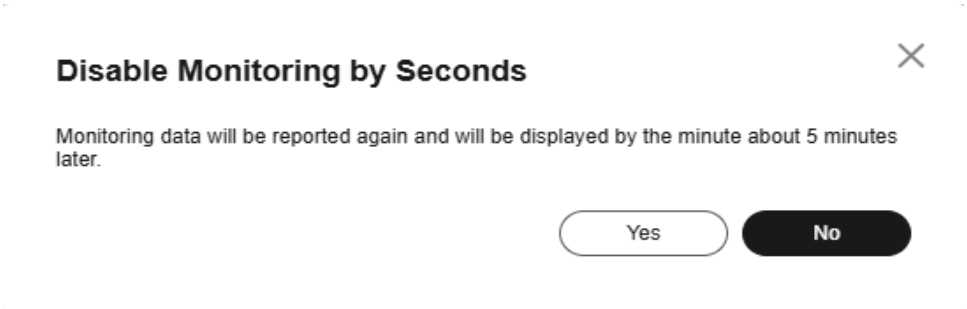
Step 5 In the navigation pane on the left, choose **Advanced O&M**.

Step 6 On the displayed page, click the **Real-Time Monitoring** tab and click next to **Monitoring by Seconds**.

Step 7 In the displayed dialog box, click **Yes**.

After you disable this function, monitoring data will be reported again and will be displayed by the minute about five minutes later.

Figure 13-2 Disable Monitoring by Seconds



----End

13.3 Viewing DDS Metrics

Scenarios

Cloud Eye monitors the statuses of DDS DB instances. You can view DDS metrics on the management console.

Monitored data takes some time before it can be displayed. The DDS status displayed on the Cloud Eye console is about 5 to 10 minutes delayed. When you create a new DDS DB instance, it takes 5 to 10 minutes before monitoring data is displayed on Cloud Eye. The monitoring data is retained for 30 days.

If you receive an alarm (for example, indicating that the data disk space is insufficient), you need to filter the instance nodes to check whether each node is normal when you view the instance monitoring data for problem location and analysis.

Prerequisites

- The DDS DB instance is running normally.

  Cloud Eye does not display the metrics of faulty or deleted DB instances or nodes. You can view the monitoring information only after the instance is restarted or recovered.
- The DB instance has been properly running for at least 10 minutes.

  For a newly created DB instance, you need to wait a bit before the monitoring metrics show up on Cloud Eye.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

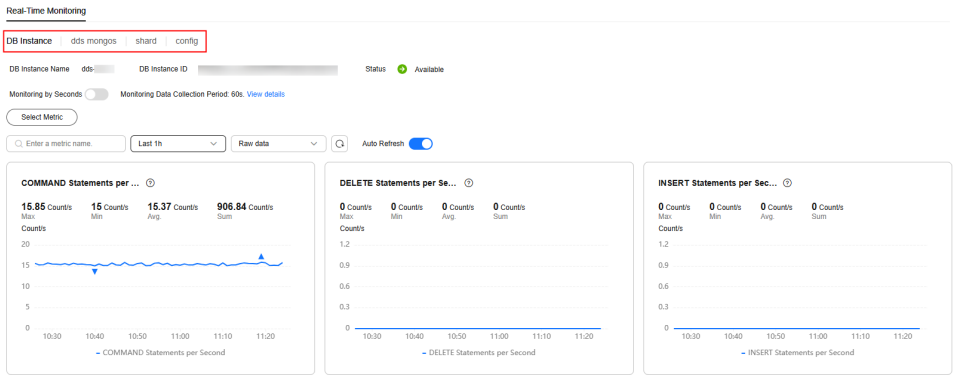
Step 4 On the **Instances** page, click the target DB instance.

Step 5 In the navigation pane on the left, choose **Advanced O&M**.

Step 6 View metrics.

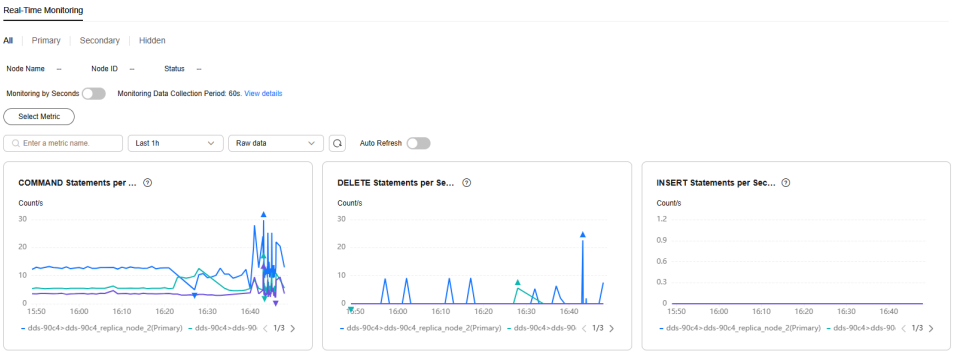
- For cluster instances, you can view metrics of instances, and dds mongos, shard, and config nodes.

  Figure 13-3 Viewing metrics of a cluster instance

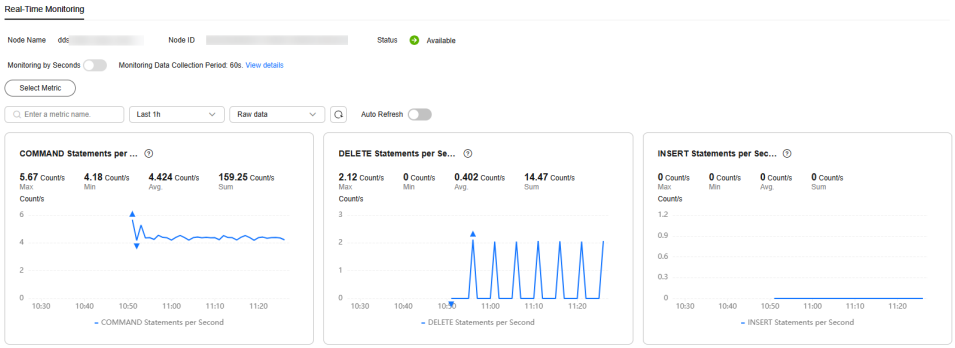
- For replica set instances, you can view metrics of the primary, secondary, and hidden nodes.

Figure 13-4 Viewing metrics of a replica set instance



- For single node instances, you can view node metrics.

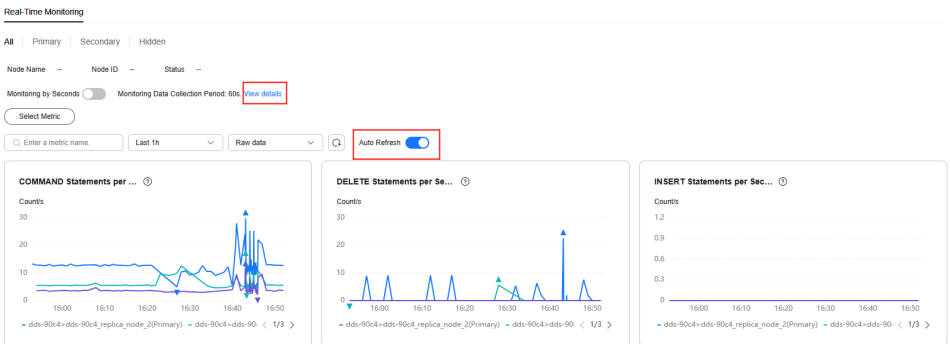
Figure 13-5 Viewing metrics of a single node instance



Step 7 View monitoring metrics of cluster instances, cluster instance nodes, and replica set instance nodes.

Step 8 In the DDS monitoring area, you can select a duration to view the monitoring data. You can view the monitoring data of the last 1 hour, 3 hours, and 12 hours.

Figure 13-6 Enabling Auto Refresh



- If automatic refresh is enabled, monitoring data is automatically refreshed every 30 seconds.
- For more metric information, click **View details** to switch to the Cloud Eye console.

----End

13.4 Configuring Alarm Rules

Scenarios

DDS allows you to set threshold rules for instance metrics. If the value of a metric exceeds the threshold, an alarm is triggered. The system automatically sends an alarm notification to the cloud account contact through SMN, helping you learn about the status of your DDS instance in a timely manner.

You can configure alarm rules on the Cloud Eye console.

Precautions

The basic alarm function is free of charge. SMN sends you the alarm messages and charges you for that. For pricing details, see .

Customizing Alarm Rules

Step 1 Log in to the management console.

Step 2 Under **Management & Governance**, click **Cloud Eye**.

Step 3 In the navigation pane on the left, choose **Alarm Management** > **Alarm Rules**.

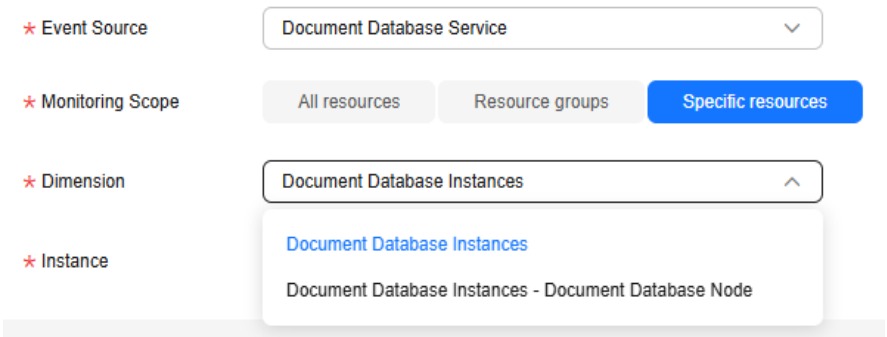
Step 4 On the displayed **Alarm Rules** page, click **Create Alarm Rule**.

Step 5 On the **Create Alarm Rule** page, follow the prompts to set the parameters.

Pay attention to the following parameters:

- **Event Source**: Select **Document Database Service**.
- **Dimension**: DDS supports instance-level and node-level monitoring dimensions. Different monitoring metrics support different monitoring dimensions. For details, see **DDS Metrics**.

Figure 13-7 Configuring monitoring dimensions



Step 6 After the alarm rule is set, the system automatically notifies you when an alarm is triggered.

----End

13.5 Managing Alarm Rules

This section describes how to enable and disable alarm reporting on the Cloud Eye console.

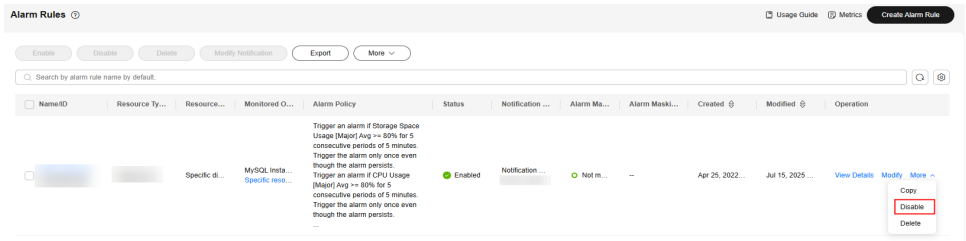
Disabling an Alarm Rule

Step 1 Log in to the management console.

Step 2 Under **Management & Governance**, click **Cloud Eye**.

Step 3 In the navigation pane on the left, choose **Alarm Management** > **Alarm Rules**, locate the alarm rule you want to disable and click **Disable** in the **Operation** column.

Figure 13-8 Disabling an Alarm Rule



Step 4 In the displayed **Disable Alarm Rule** dialog box, click **Yes** to disable the alarm rule.

If you want to disable multiple alarm rules, on the **Alarm Rules** page, select multiple alarm rules, and click **Disable** in the upper left of the alarm rule list. In the displayed **Disable Alarm Rule** dialog box, click **Yes**.

----End

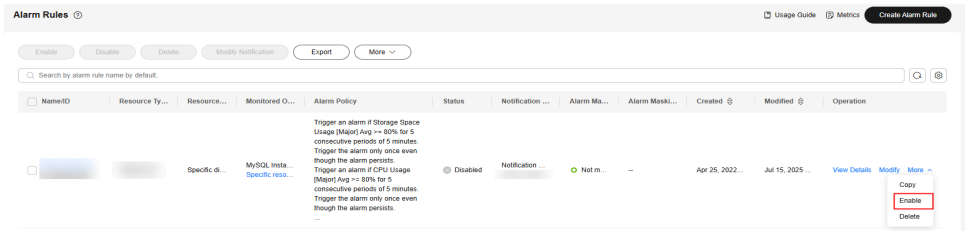
Enabling an Alarm Rule

Step 1 Log in to the management console.

Step 2 Under **Management & Governance**, click **Cloud Eye**.

Step 3 In the navigation pane on the left, choose **Alarm Management** > **Alarm Rules**, locate the alarm rule you want to enable and click **Enable** in the **Operation** column.

Figure 13-9 Enabling an Alarm Rule



Step 4 In the displayed **Enable Alarm Rule** dialog box, click **Yes** to enable the alarm rule.

If you want to enable multiple alarm rules, on the **Alarm Rules** page, select multiple alarm rules, and click **Enable** in the upper left of the alarm rule list. In the displayed **Enable Alarm Rule** dialog box, click **Yes**.

----End

14 Logs

14.1 Log Reporting

Prerequisites

You have created a log group and a log stream on the Log Tank Service (LTS) console.

Scenarios

If you enable log reporting to LTS, new audit logs, error logs, and slow query logs generated for DDS DB instances will be uploaded to LTS for management. You can view details about audit logs, error logs, and slow query logs of DDS DB instances, including searching for logs, visualizing logs, downloading logs, and viewing real-time logs.

The following operations use audit logs as an example:

- Enable log reporting to LTS for a single DB instance by referring to **Enabling Log Reporting to LTS for a Single DB Instance**.
- Edit log reporting to LTS for a single DB instance by referring to **Editing Log Reporting to LTS for a Single DB Instance**.
- Disable log reporting to LTS for a single DB instance by referring to **Disabling Log Reporting to LTS for a Single DB Instance**.
- Enable log reporting to LTS in batches by referring to **Enabling Log Reporting to LTS in Batches**.
- Disable log reporting to LTS in batches by referring to **Disabling Log Reporting to LTS in Batches**.

Precautions

- Logs record all requests sent to your DB instance and are stored in LTS.
- This request does not take effect immediately. There is a delay of about 10 minutes.
- After this function is enabled, all audit policies are reported by default.

- Audit logs are generated every hour. If the size of an audit log exceeds 10 MB, a new audit log is generated.
- If **Audit Policy** is enabled, LTS reuses the audit policy set for your DB instance and you will also be billed for reporting audit logs to LTS. (Only after you disable **Audit Policy**, the fee will be terminated.)
- If you enable audit log reporting to LTS for an instance with the **Audit Policy** toggle switch turned on, you can turn off this switch only when the instance status becomes available.

Enabling Log Reporting to LTS for a Single DB Instance

Step 1 Log in to the console.

Step 2 Click in the upper left corner and select a region and a project.

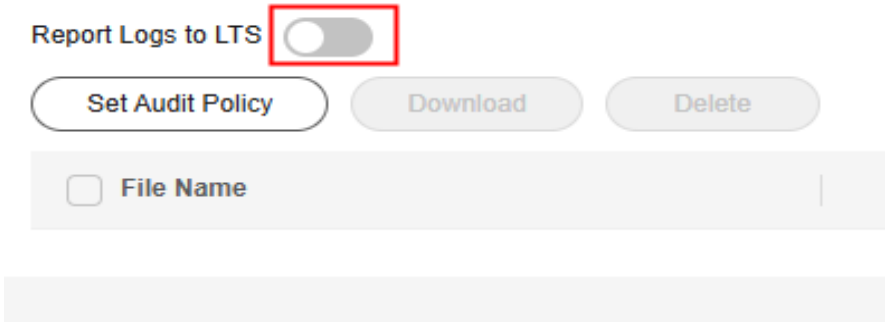
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate a target DB instance and click its name.

Step 5 In the navigation pane on the left, choose **Audit Logs**.

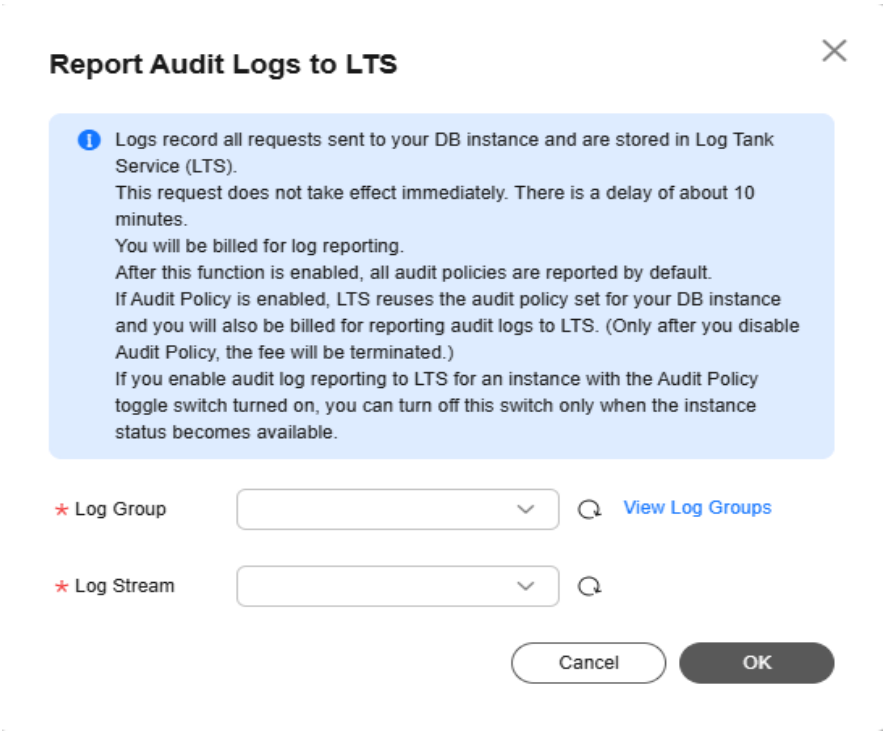
Step 6 On the **Audit Logs** page, click next to **Report Logs to LTS**.

Figure 14-1 Enabling Report Logs to LTS



Step 7 In the displayed dialog box, specify **Log Group** and **Log Stream**.

Figure 14-2 Enabling audit log reporting to LTS



NOTE

Step 8 Click **OK**.

----End

Editing Log Reporting to LTS for a Single DB Instance

Step 1 Log in to the console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate a target DB instance and click its name.

Step 5 In the navigation pane on the left, choose **Audit Logs**.

Step 6 On the **Audit Logs** page, click **Edit** next to the **Report Logs to LTS** toggle switch.

NOTE

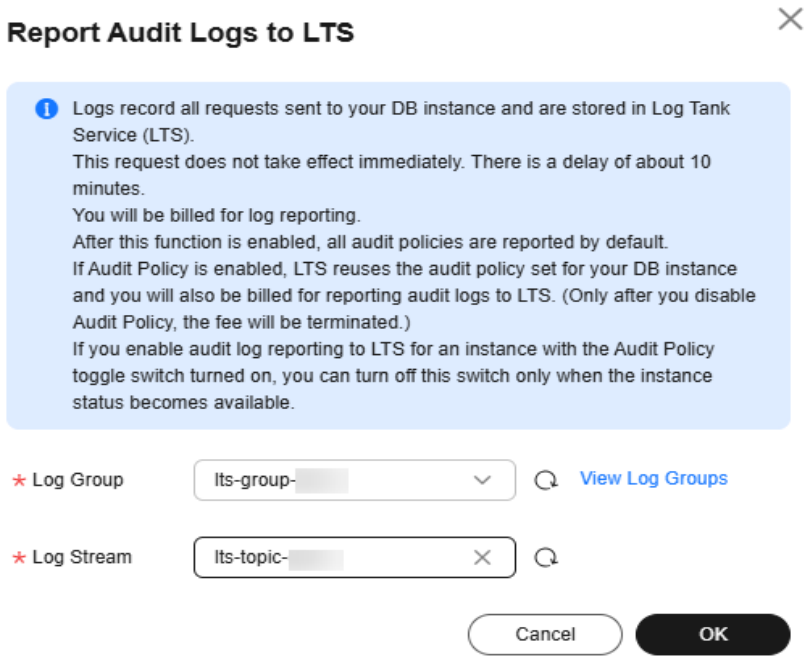
The editing function is available only when the **Report Logs to LTS** toggle switch is turned on.

Step 7 In the displayed dialog box, specify **Log Group** and **Log Stream**.

NOTE

Select the target log group and log stream.

Figure 14-3 Editing audit log reporting to LTS



Step 8 Click **OK**.

----End

Disabling Log Reporting to LTS for a Single DB Instance

Step 1 Log in to the console.

Step 2 Click in the upper left corner and select a region and a project.

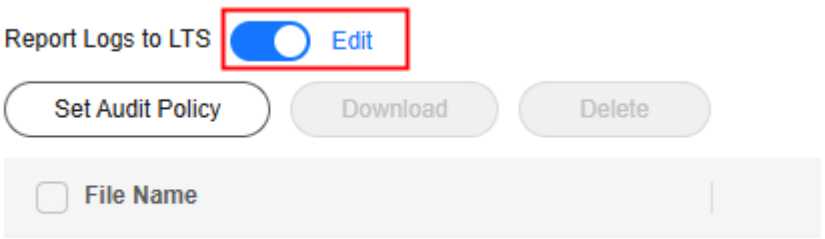
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate a target DB instance and click its name.

Step 5 In the navigation pane on the left, choose **Audit Logs**.

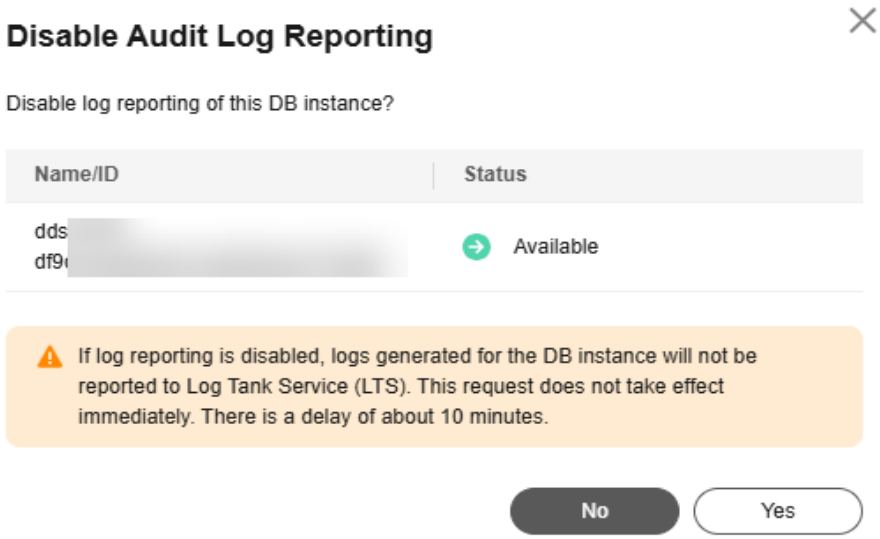
Step 6 On the **Audit Logs** page, click next to **Report Logs to LTS**.

Figure 14-4 Disabling Report Logs to LTS



Step 7 In the displayed dialog box, click **Yes**.

Figure 14-5 Disabling audit log reporting to LTS



----End

Enabling Log Reporting to LTS in Batches

Step 1 Log in to the console.

Step 2 Click in the upper left corner and select a region and a project.

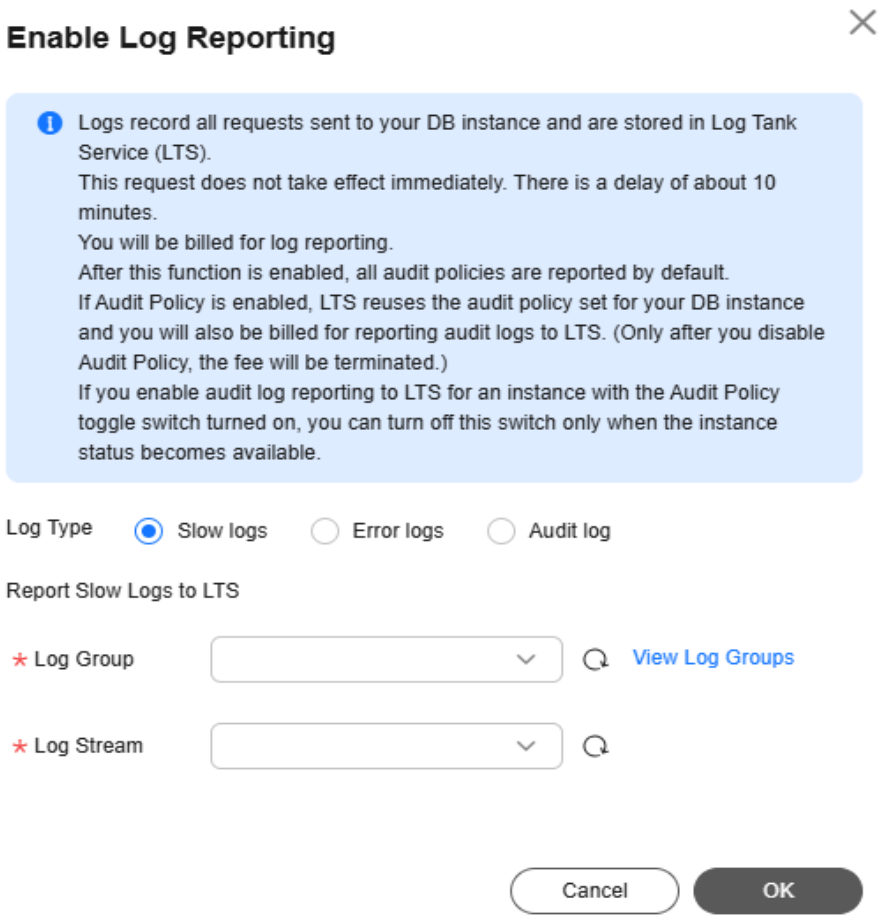
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Log Reporting**.

Step 5 Select target DB instances and click **Enable Log Reporting**.

Step 6 In the displayed dialog box, specify **Log Group** and **Log Stream**.

Figure 14-6 Enabling log reporting to LTS in batches



NOTE

- Select the target log group and log stream.

Step 7 Click **OK**.

----End

Disabling Log Reporting to LTS in Batches

Step 1 Log in to the console.

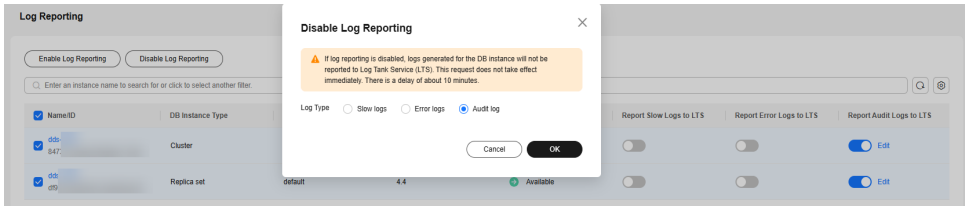
Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, choose **Log Reporting**.

Step 5 Select target DB instances and click **Disable Log Reporting**.

Figure 14-7 Disabling log reporting to LTS in batches



Step 6 In the displayed dialog box, click **OK**.

----End

14.2 Error Logs

14.2.1 Viewing Error Logs on the LTS Console

You can analyze, search for, monitor, download, and view real-time logs on the LTS console.

Querying Error Logs Reported to LTS

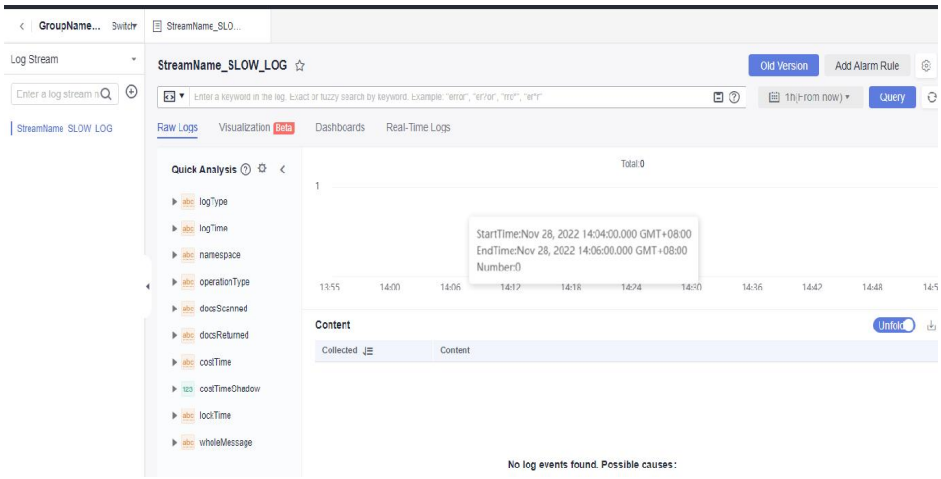
NOTE

You have enabled log reporting to LTS. For details, see **Log Reporting**.

Step 1 Click ☰ in the upper left corner of the page and choose **Management & Governance** > **Log Tank Service**.

Step 2 In the **Log Groups** area, locate a target log group and click its name.

Figure 14-8 Viewing log details



----End

Downloading Error Logs Reported to LTS

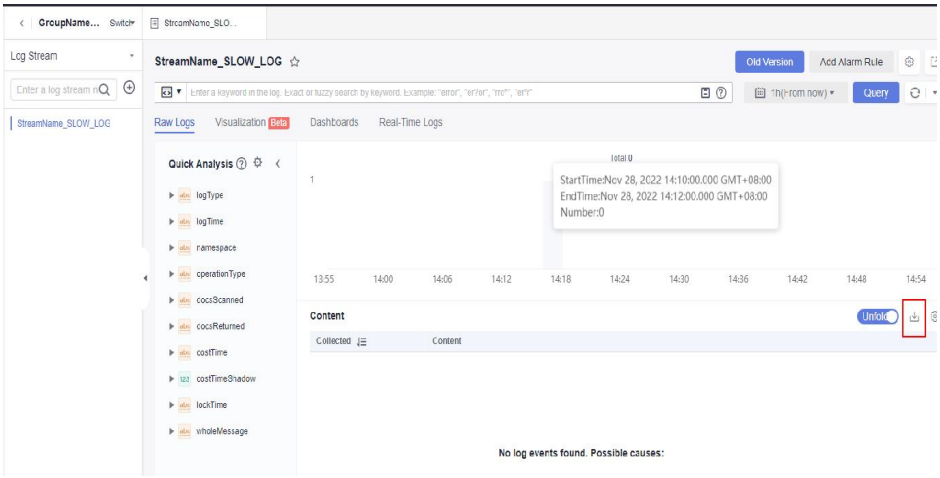
NOTE

If you have enabled log reporting to LTS for your DB instance in **Log Reporting**, you can download logs on the LTS console.

Step 1 Click ☰ in the upper left corner of the page and choose **Management & Governance** > **Log Tank Service**.

Step 2 In the **Log Groups** area, locate a target log group and click its name.

Figure 14-9 Downloading logs



Step 3 Click ⬇.

----End

14.2.2 Viewing Error Logs on the DDS Console

DDS log management allows you to view database-level logs, including warning- and error-level logs generated during database running, which help you analyze system problems.

Viewing and Exporting Log Details

Step 1 Log in to the management console.

Step 2 Click 📍 in the upper left corner and select a region and a project.

Step 3 Click ☰ in the upper left corner of the page and choose **Databases** > **Document Database Service**.

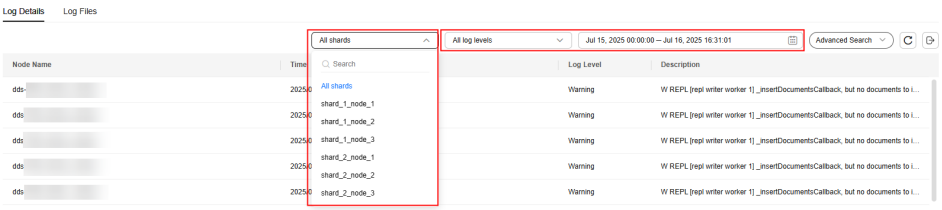
Step 4 On the **Instances** page, click the instance name.

Step 5 In the navigation pane on the left, choose **Error Logs**.

Step 6 On the displayed page, click **Error Logs**. Then, view the log details on the **Log Details** tab.

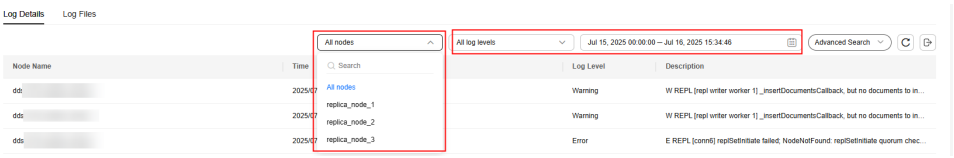
- For a cluster instance, you can view error logs of the dds mongos, shard, and config nodes.

  Figure 14-10 Viewing error logs of a cluster instance

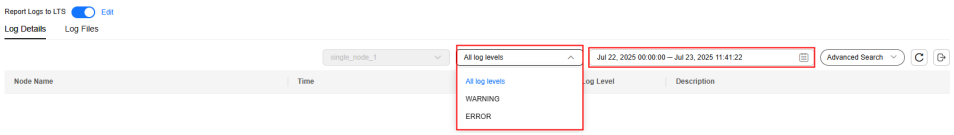
- For a replica set instance, you can view the error logs of the primary, secondary, hidden nodes, and read replicas.

  Figure 14-11 Viewing error logs of a replica set instance

- For a single node instance, you can view error logs of the current node.

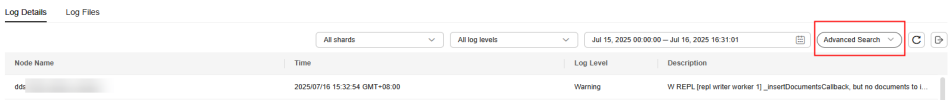
  Figure 14-12 Viewing error logs of a single-node instance

- You can view up to 2,000 error logs of a specified node type, at a specified level, and within a specified period.

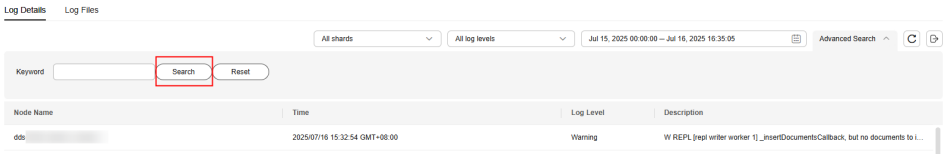
Step 7 On the **Log Details** tab, click **Advanced Search**.

Figure 14-13 Advanced search



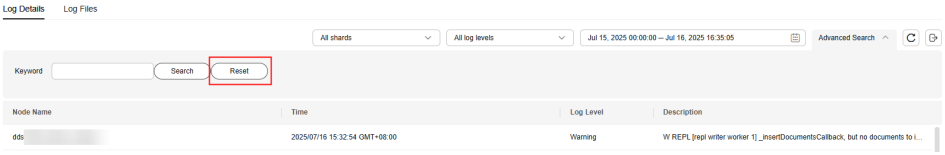
Step 8 Specify **Keyword** and click **Search** to view log information.

Figure 14-14 Setting advanced search parameters



Step 9 To clear the parameter settings of **Advanced Search**, click **Reset**.

Figure 14-15 Resetting advanced search parameters



Step 10 On the **Log Details** tab, click in the upper right corner of the log list to export log details.

- View the .csv file exported to your local PC.
- Up to 2,000 log details can be exported at a time.

----End

Downloading Logs

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

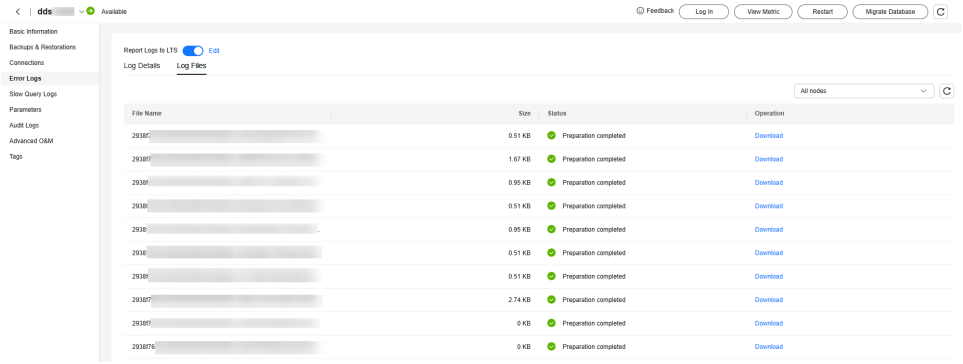
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the Community Edition instance name.

Step 5 In the navigation pane on the left, choose **Error Logs**.

Step 6 On the **Error Logs** page, click the **Log Files** tab. Locate a log whose status is **Preparation completed** and click **Download** in the **Operation** column.

Figure 14-16 Error Logs



- The system automatically loads the downloading preparation tasks. The time it takes to download the logs depends on the file size and on the network environment.
  - During the downloading preparation, the log status is **Preparing**.
  - Once the logs are ready for download, the log status changes to **Preparation completed**.
  - If the downloading preparation fails, the log status is **Abnormal**.
- You can download only one log file from a node. The maximum size of a log file to be downloaded is 40 MB.

- The download link is valid for 15 minutes. After the download link expires, a message is displayed indicating that the download link has expired. To download the log, click **OK**.

----End

14.3 Slow Query Logs

14.3.1 Viewing Slow Query Logs on the LTS Console

You can analyze, search for, monitor, download, and view real-time logs on the LTS console.

Querying Slow Query Logs Reported to LTS

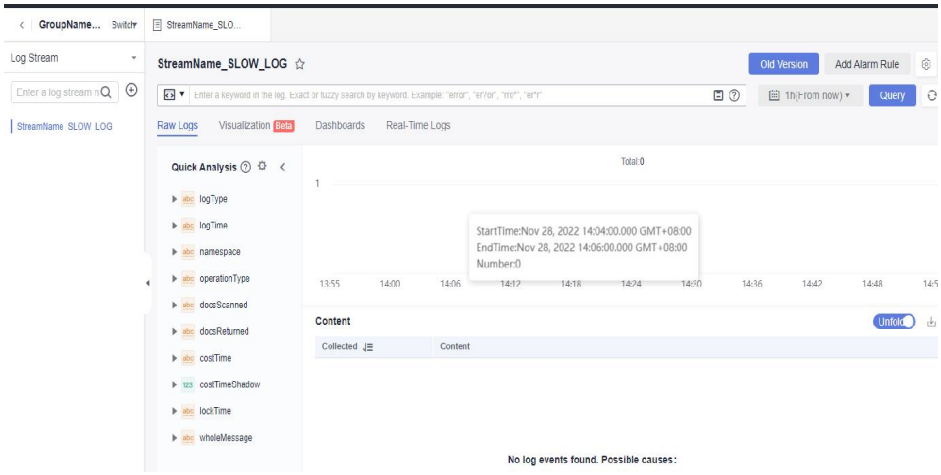
NOTE

You have enabled log reporting to LTS. For details, see **Log Reporting**.

Step 1 Click ☰ in the upper left corner of the page and choose **Management & Governance** > **Log Tank Service**.

Step 2 In the **Log Groups** area, locate a target log group and click its name.

Figure 14-17 Viewing log details



----End

Downloading Slow Query Logs Reported to LTS

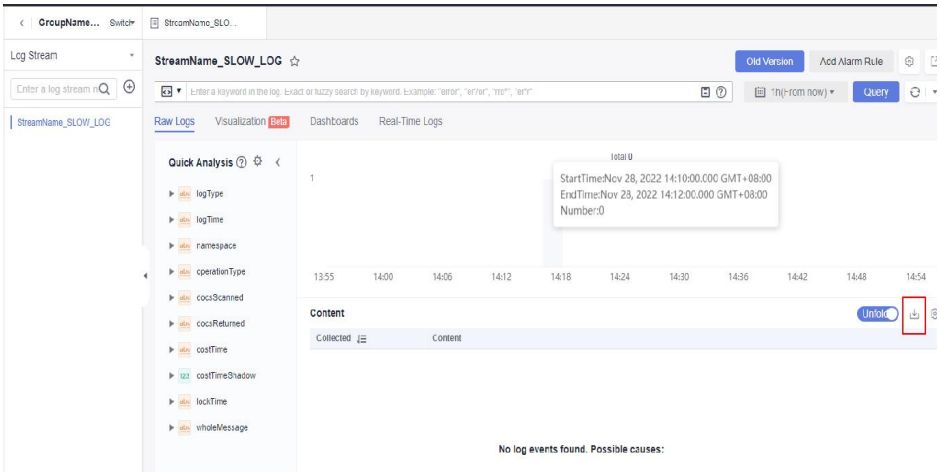
NOTE

If you have enabled log reporting to LTS for your DB instance in **Log Reporting**, you can download logs on the LTS console.

Step 1 Click ☰ in the upper left corner of the page and choose **Management & Governance** > **Log Tank Service**.

Step 2 In the **Log Groups** area, locate a target log group and click its name.

Figure 14-18 Downloading logs



Step 3 Click .

----End

14.3.2 Viewing Slow Query Logs on the DDS Console

Slow query logs record statements that exceed **operationProfiling.slowOpThresholdMs** (500 seconds by default). You can view log details and statistics to identify statements that are executing slowly and optimize the statements. You can also download slow query logs for service analysis.

Precautions

- Community Edition instances allow you to view and export log details, enable Show Original Log, and download log files on the management console.
- The Show Original Log function cannot be enabled when you delete DB instances, add nodes, change DB instance class, rebuild secondary node, or the DB instance is frozen.
- If **Show Original Log** is being enabled, you cannot delete instances, add nodes, or change instance class.
- By default, if the execution time of a SQL statement exceeds 500 ms, a slow query log is recorded.
- When the size of slow query logs reaches a specified threshold, old data is automatically deleted. If you need to analyze slow query logs, download the logs on the console in a timely manner.
- You can query slow logs for the last 30 days.
- You cannot delete slow query logs of DDS.
- When you export data on the **Log Details** page, all logs displayed on the current page will be exported.
- For details about how to sort slow query logs by field, such as execution completion time, SQL statement, client IP address, user, execution duration,

lock wait time, scanned documents, returned documents, and scanned indexes, see "Slow Query Logs" in *Data Admin Service (DAS) User Guide*.

- Slow query logs may have a delay of several seconds to minutes, depending on the number of generated slow query logs and the DB instance load.
- In slow query log monitoring, the data of each monitored node is generated based on the total number of slow query logs generated 5 minutes before the time point.
- Slow query logs do not have strict consistency. That is, the slow query logs displayed on the page may not include the full slow query logs. The system collects slow query logs periodically. If slow query logs are generated too frequently, the collection period may not cover all slow query logs.

Parameter description

Table 14-1 Parameters related to DDS slow query logs

Parameter	Description
operationProfiling.slowOpThresholdMs	Queries that exceed the threshold in the unit of ms are deemed slow. The default value is **500 ms**. Unless otherwise specified, keeping the default value is recommended.

Enabling Show Original Log

NOTE

- If **Show Original Log** is enabled, original logs are displayed. By default, the system automatically deletes original logs after 30 days, and the period cannot be changed.
- If the instance a slow query log belongs to is deleted, related logs are deleted along with it.
- **Show Original Log** can be disabled after it is enabled. The slow query logs reported before the function is disabled are displayed. The slow query logs reported after the function is disabled are not displayed.

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

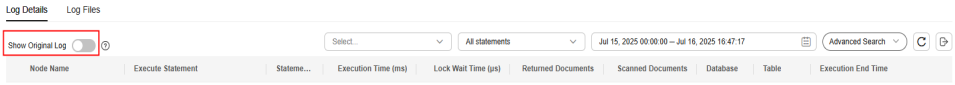
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the navigation pane on the left, choose **Slow Query Logs**.

Step 6 On the displayed page, click **Slow Query Logs**. Then, click on the **Log Details** tab.

Figure 14-19 Enabling Show Original Log



Step 7 In the displayed dialog box, click **Yes** to enable the function of slowing original logs.

----End

Viewing and Exporting Log Details

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

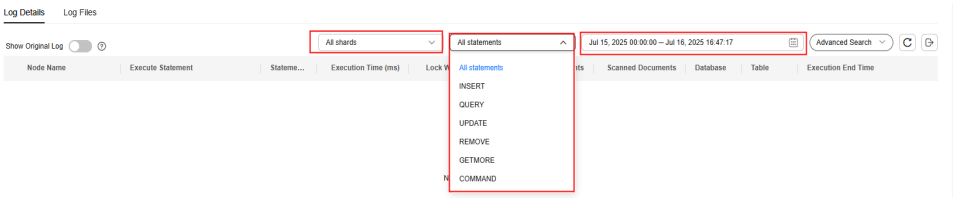
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the navigation pane on the left, choose **Slow Query Logs**.

Step 6 On the **Slow Query Logs** page, set search criteria on the **Log Details** tab to view log information.

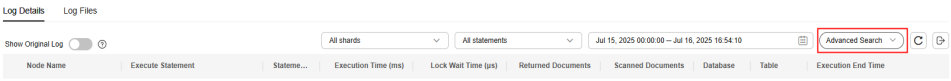
Figure 14-20 Querying slow query logs



- Log records of all shards of a cluster instance
- Log records of all nodes in a replica set instance
- Slow query logs of a node in different time periods
- Slow query statements of the following levels
  - All statement type
  - INSERT
  - QUERY
  - UPDATE
  - REMOVE
  - GETMORE
  - COMMAND
- You can view up to 2,000 slow logs of a specified node type, at a specified level, and within a specified period.

Step 7 On the **Log Details** tab, click **Advanced Search**.

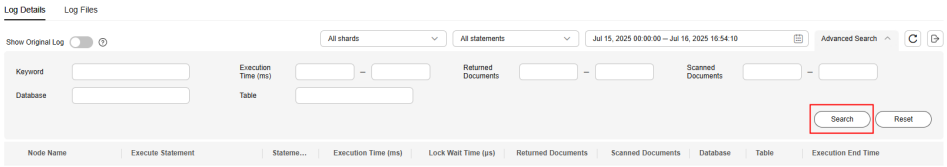
Figure 14-21 Advanced search



NOTE

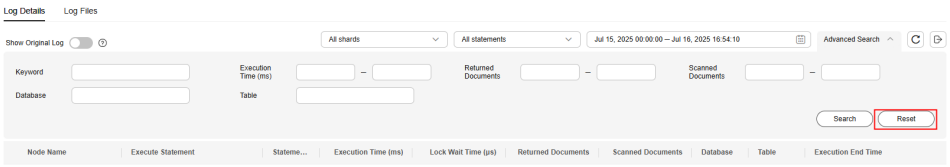
Step 8 Specify **Keyword**, **Execution Time (ms)**, **Returned Documents**, **Scanned Documents**, **Database**, **Table**, **Scanned Indexes**, **Username**, and **Client IP Address** and click **Search** to view log information.

Figure 14-22 Setting advanced search parameters



Step 9 To clear the parameter settings of **Advanced Search**, click **Reset**.

Figure 14-23 Resetting advanced search parameters



Step 10 On the **Log Details** tab, click in the upper right corner of the log list to export log details.

- View the .csv file exported to your local PC.
- Up to 2,000 log details can be exported at a time.

----End

Downloading Logs

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

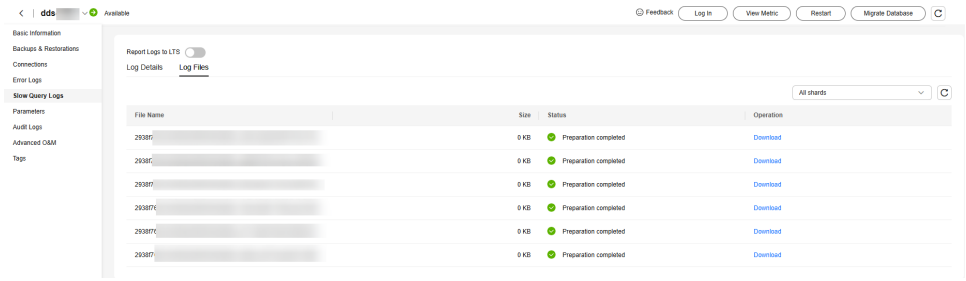
Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the navigation pane on the left, choose **Slow Query Logs**.

Step 6 On the **Slow Query Logs** page, click the **Log Files** tab. Locate a log whose status is **Preparation completed** and click **Download** in the **Operation** column.

Figure 14-24 Slow Query Logs



- The system automatically loads the downloading preparation tasks. The time required depends on the log file size and the network environment.
  - During the downloading preparation, the log status is **Preparing**.
  - Once the logs are ready for download, the log status changes to **Preparation completed**.
  - If the downloading preparation fails, the log status is **Abnormal**.
- You can download only one log file from a node. The maximum size of a log file to be downloaded is 40 MB.
- The download link is valid for 5 minutes. After the download link expires, a message is displayed indicating that the download link has expired. To download the log, click **OK**.

----End

Reference

How Do I Optimize Slow Operations?

14.4 Audit Logs

14.4.1 Audit Log Policy Management

Scenarios

An audit log records operations performed on your databases and collections. The generated log files are stored in OBS. Auditing logs can enhance your database security and help you analyze the cause of failed operations.

Precautions

- The audit policy of a DDS DB instance is disabled by default. You can enable it based on your service requirements. After the function is enabled, the system records audit information about read and write operations. The impacts on database performance vary depending on the service model. If the database encounters a performance bottleneck after this function is enabled, you need to temporarily disable the function or upgrade the specifications.
- DDS checks generated audit logs. If the retention period of logs exceeds the period you set, DDS will delete the logs. It is recommended that audit logs be stored for more than 180 days for tracing and problem analysis.

- After the audit policy is modified, DDS audits logs according to the new policy and the retention period of the original audit logs is subject to the modified retention period.
- By default, audit logs are generated every hour. If the size of an audit log exceeds 20 MB, a new audit log is generated.
- Your data must be encoded in UTF-8 format. For data in other format, the auditing result of the corresponding statement may be missing or contain garbled characters.
- Audit log files stored on OBS are invisible to you. They are only visible in the DDS backend management system.

Example Traces

The following is an example of querying the replica set status. For details about the fields, see **Table 1**.


```
{
 "atype": "query",
 "ts": {
  "$date": "2025-02-27T12:35:40.512+0000"
 },
 "local": {
   "ip": "192.168.30.90",
   "port": 8635
 },
 "remote": {
  "ip": "192.168.26.20",
  "port": 43818
 },
 "users": [
  {
   "user": "rwuser",
   "db": "admin"
  }
 ],
 "roles": [
  {
   "role": "root",
   "db": "admin"
  }
 ],
 "param": {
  "command": "query",
  "ns": "test.a",
  "args": {
   "find": "a",
   "filter": {
   },
   "lsid": {
    "id": {
     "$binary": "7kwKmFh9TxGStjtn5Urt2w==",
     "$type": "04"
    }
   },
   "$clusterTime": {
    "clusterTime": {
     "$timestamp": {
      "t": 1740659738,
      "i": 2
```



```
      }
     },
      "signature": {
       "hash": {
         "$binary": "t2eFI3gB9ZruNYf5Y0FIgAZRCr4=",
         "$type": "00"
       },
       "keyId": {
          "$numberLong": "7472645695153897501"
       }
      }
    },
    "$db": "test"
  }
 },
 "result": 0
 "sessionId": "1641365",
 "username": "rwuser",
 "clientname": "MongoDB Shell",
 "affectRows": "0",
 "startTime": "1740659740512716",
 "endTime": "1740659740513937",
 "docsExamined": "0",
 "keysExamined": "0",
 "returnNum": "0",
 "success": true
}
```


NOTE

New fields have been added to audit logs in versions later than 250130. You can download the original logs to view the new fields. The audit log fields displayed on the console remain unchanged.

Table 14-2 Fields

Field	Type	Description
atype	stringData	Operation type of the event.
ts	Date_t	Timestamp when the event is generated.
local(ip)	BSONObj (ip + port)	Local IP address.
remote(ip)	BSONObj (ip + port)	Remote IP address.
users	BSONArray	User group.
roles	BSONArray	Role group.
param	BSONObj	Parameters involved in the command.
result	ErrorCodes::Error	Permission authentication result (not the command execution result).

Field	Type	Description
sessionId	stringData	Session ID, which uniquely identifies a connection.
username	stringData	User who runs the command.
clientname	stringData	Information about the client that runs the command.
affectRows	long long	Number of rows in the document that is affected by the command execution.
startTime	timestamp	Time when the command starts to be executed.
endTime	timestamp	Time when the command ends to be executed.
docsExamined	long long	Number of documents read during command execution.
keysExamined	long long	Number of indexes read during command execution.
returnNum	long long	Number of documents in the command execution result.
success	bool	Whether the command execution status is normal.

Configuring the Audit Policy

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the navigation pane on the left, choose **Audit Logs**.

Step 6 On the **Audit Logs** page, click **Set Audit Policy**.

Step 7 On the displayed page, click  .

Step 8 Configure required parameters and click **OK** to enable the audit policy.

Figure 14-25 Enabling audit policy

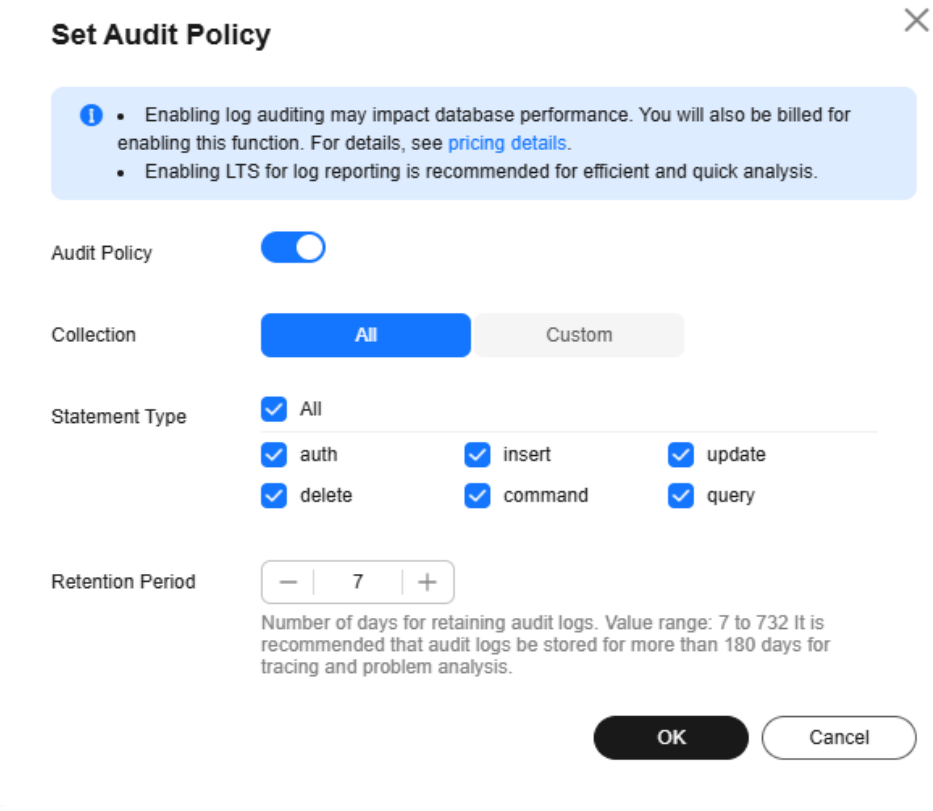


Table 14-3 Parameter description

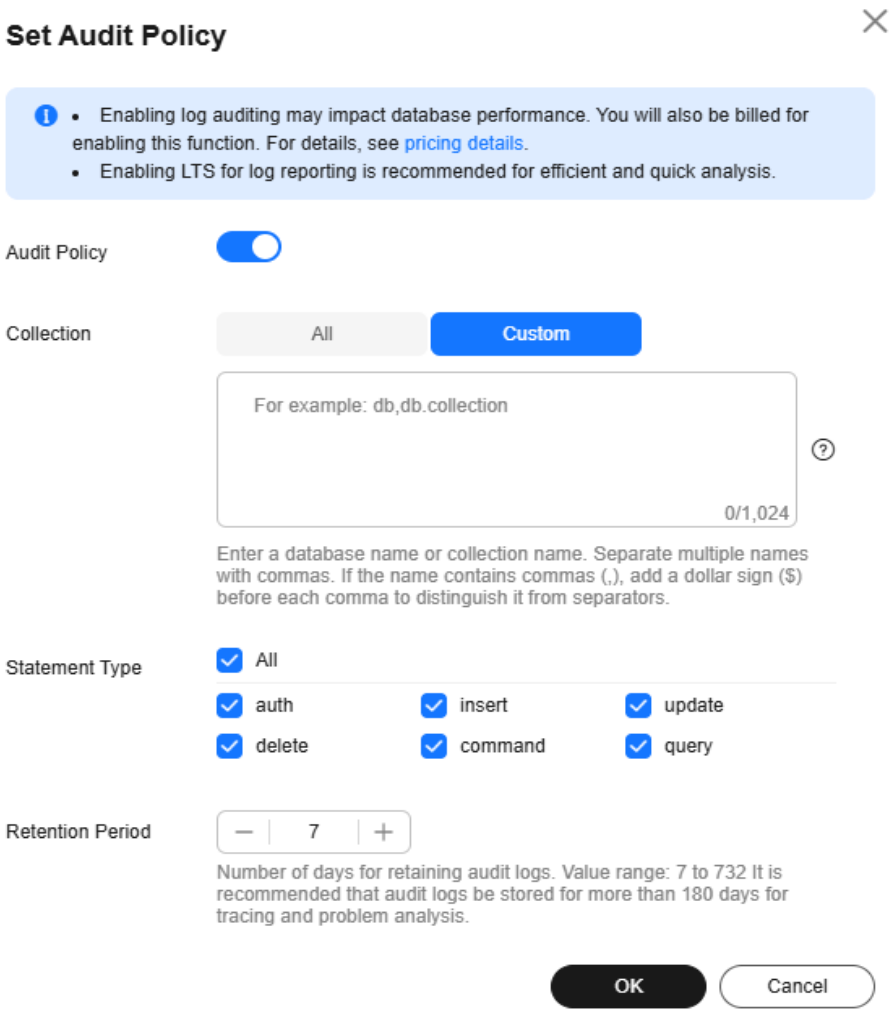
Parameter	Description
All	Audit all collections in the instance.
Custom	Audit specified databases or collections in the instance. The database or collection name cannot contain spaces or the following special characters: /\' : "[]{}() The dollar sign ($) can be used only as an escape character. The database name can contain a maximum of 64 characters. If you enter a combined database and collection name, the total name length is 120 characters with the database name length of no more than 64 characters and the collection name cannot be blank, contain **null**, or use **system.** in prefix.

Parameter	Description
Statement Type	You can query audit logs of specified statements in a collection, including auth, insert, update, delete, command and query statements.
Retention Days	The number of days to retain audit logs. Range: 7 to 732

- After the audit policy is enabled, you can modify it as required. After the modification, logs are generated according to the new policy and the retention period of the original logs is subject to the modified retention period.

  To modify the audit policy, click **Set Audit Policy**. In the dialog box that is displayed, modify the audit policy.

  Figure 14-26 Modifying the audit policy



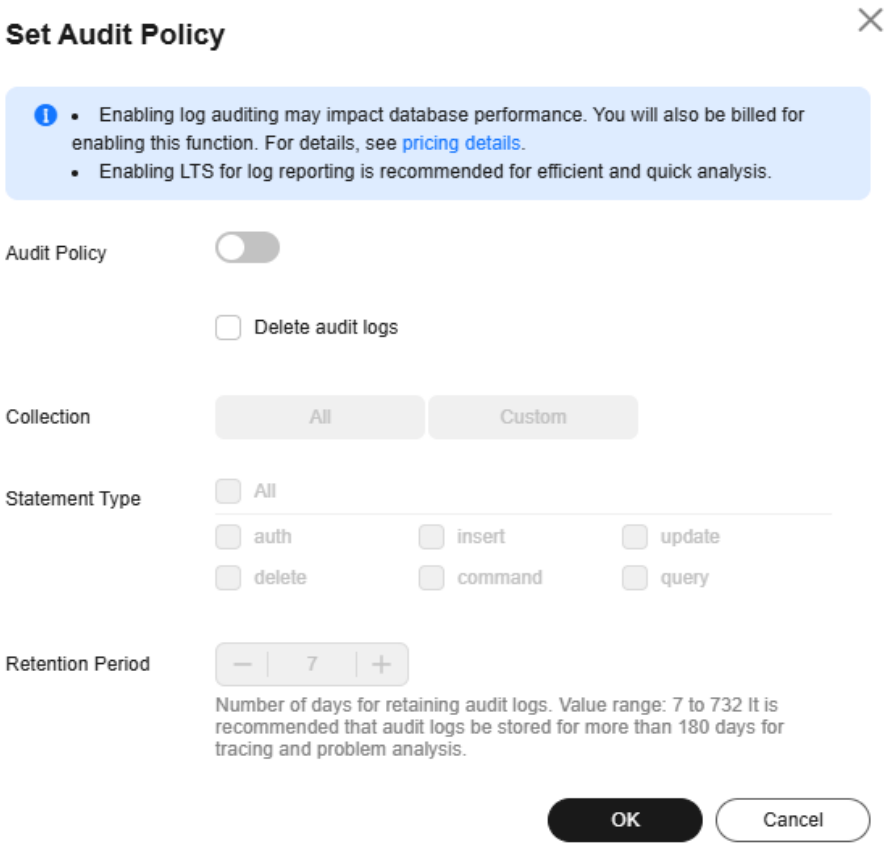
- Disable the audit policy.

NOTE

After the audit policy is disabled, no audit log is generated.

To disable the audit policy, click  . **Figure 14-27** shows the dialog box for setting the backup policy.

Figure 14-27 Disabling audit policy



You can determine whether to delete all audit logs:

- If you do not select **Delete audit logs**, all audit logs within the retention period will be retained. You can manually delete them later.
- If you select **Delete audit logs**, all audit logs within the retention period will be deleted.

Click **OK**.

----End

14.4.2 Viewing Audit Logs on the LTS Console

You can analyze, search for, monitor, download, and view real-time logs on the LTS console.

Querying Audit Logs Reported to LTS

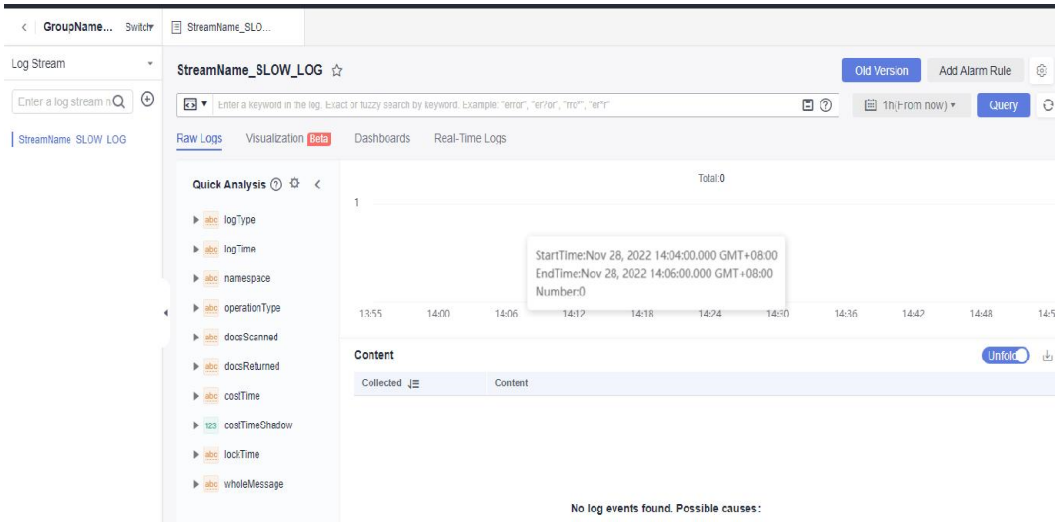
NOTE

You have enabled log reporting to LTS. For details, see **Log Reporting**.

Step 1 Click ☰ in the upper left corner of the page and choose **Management & Governance** > **Log Tank Service**.

Step 2 In the **Log Groups** area, locate a target log group and click its name.

Figure 14-28 Viewing log details



----End

Downloading Audit Logs Reported to LTS

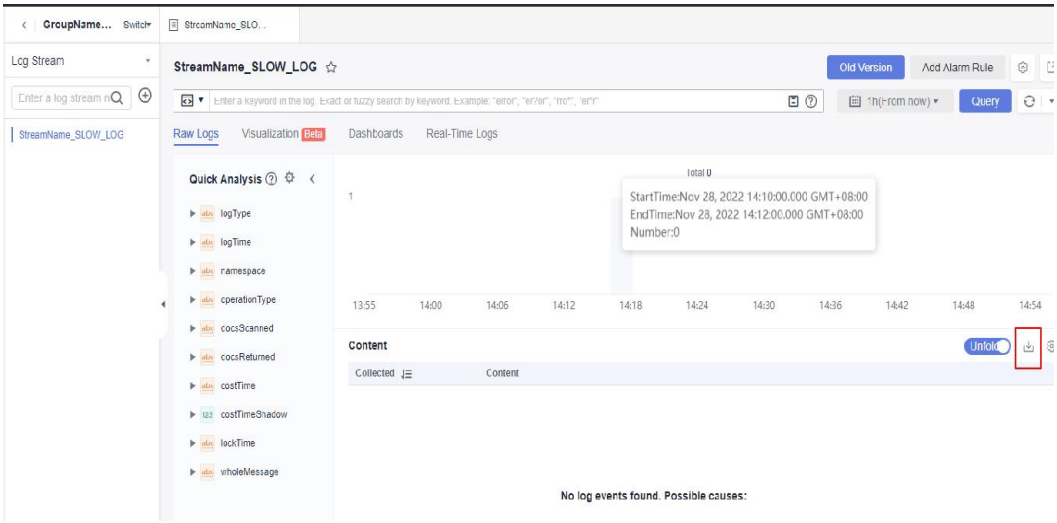
NOTE

If you have enabled log reporting to LTS for your DB instance in **Log Reporting**, you can download logs on the LTS console.

Step 1 Click ☰ in the upper left corner of the page and choose **Management & Governance** > **Log Tank Service**.

Step 2 In the **Log Groups** area, locate a target log group and click its name.

Figure 14-29 Downloading logs



Step 3 Click .

----End

14.4.3 Viewing Audit Logs on the DDS Console

You can view, download, and delete audit logs on the DDS console.

Querying Audit Logs

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate a target DB instance and click its name.

Step 5 In the navigation pane on the left, choose **Audit Logs**.

Step 6 On the **Audit Logs** page, locate a target log file and click **Download** in the **Operation** column to download the log file to the local PC for query.

----End

Downloading Logs

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate a target DB instance and click its name.

Step 5 In the navigation pane on the left, choose **Audit Logs**.

Step 6 On the **Audit Logs** page, locate a target log file and click **Download** in the **Operation** column.

- The system automatically loads the downloading preparation tasks. The time required depends on the log file size and the network environment.
- The download link is valid for 5 minutes. After the download link expires, a message is displayed indicating that the download link has expired. To download the log, click **OK**.

  NOTE

  You are advised to download no more than six audit log files at a time. Too many files can fail to be downloaded completely due to the limit on the number of concurrent requests of the browser.

----End

Deleting Logs

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate a target DB instance and click its name.

Step 5 In the navigation pane on the left, choose **Audit Logs**.

Step 6 On the **Audit Logs** page, locate a target log file and click **Delete** in the **Operation** column.

Step 7 Click **Yes**.

----End

15 CTS Auditing

15.1 Key Operations Recorded by CTS

With Cloud Trace Service (CTS), you can record operations associated with DDS for later query, audit, and backtrack operations.

Table 15-1 Key operations on DDS

Operation	Resource	Trace Name
Restoring data to a new DB instance	instance	ddsRestoreToNewInstance
Restoring to an existing DB instance	instance	ddsRestoreToOldInstance
Creating a DB instance	instance	ddsCreateInstance
Deleting a DB instance	instance	ddsDeleteInstance
Restarting a DB instance	instance	ddsRestartInstance
Scaling up a DB instance	instance	ddsGrowInstance
Scaling up storage space	instance	ddsExtendInstanceVolume
Resetting the database password	instance	ddsResetPassword
Renaming a DB instance	instance	ddsRenameInstance
Switching SSL	instance	ddsSwitchSsl
Modifying a DB instance port	instance	ddsModifyInstancePort
Creating a backup	backup	ddsCreateBackup
Deleting a backup	backup	ddsDeleteBackup

Operation	Resource	Trace Name
Setting a backup policy	backup	ddsSetBackupPolicy
Applying a parameter template	parameterGroup	ddsApplyConfigurations
Replicating a parameter template	parameterGroup	ddsCopyConfigurations
Resetting a parameter template	parameterGroup	ddsResetConfigurations
Creating a parameter template	parameterGroup	ddsCreateConfigurations
Deleting a parameter template	parameterGroup	ddsDeleteConfigurations
Updating a parameter template	parameterGroup	ddsUpdateConfigurations
Binding an EIP	instance	ddsBindEIP
Unbinding an EIP	instance	ddsUnBindEIP
Editing a tag	tag	ddsModifyTag
Deleting an instance tag	tag	ddsDeleteInstanceTag
Adding an instance tag	tag	ddsAddInstanceTag
Rolling back upon scaling-up failure	instance	ddsDeleteExtendedDdsNode
Changing DB instance classes	instance	ddsResizeInstance
Unfreezing a DB instance	instance	ddsUnfreezeInstance
Freezing a DB instance	instance	ddsFreezeInstance
Changing a private IP address	instance	ddsModifyIP
Modifying a private domain name	instance	ddsModifyDNSName
Enabling or disabling cluster balancing	instance	ddsSetBalancer
Switching the internal communication mode	instance	ddsSwitchInnerSsl
Adding read replicas	instance	AddReadonlyNode
Enabling shard/config IP address for a cluster instance	instance	ddsCreateIp

Operation	Resource	Trace Name
Changing a security group	instance	ddsModifySecurityGroup
Changing an AZ	instance	ddsMigrateAvailabilityZone
Modifying instance remarks	instance	ddsModifyInstanceRemark
Configuring a maintenance window	instance	ddsModifyInstanceMaintenanceWindow
Upgrading patches	instance	ddsUpgradeDatastorePatch
Performing a primary/standby switchover	instance	ddsReplicaSetSwitchover
Configuring cross-CIDR access	instance	ddsModifyInstanceSourceSubnet
Modifying instance parameters	parameterGroup	ddsUpdateInstanceConfigurations
Exporting a parameter template for a DB instance	parameterGroup	ddsSaveConfigurations
Setting a cross-region backup policy	backup	ddsModifyOffsiteBackupPolicy
Enabling plaintext display of slow query logs	instance	ddsOpenSlowLogPlaintextSwitch
Disabling plaintext display of slow query logs	instance	ddsCloseSlowLogPlaintextSwitch
Downloading error or slow query logs	instance	ddsDownloadLog
Enabling the audit policy for a DB instance	instance	ddsOpenAuditLog
Disabling the audit policy for a DB instance	instance	ddsCloseAuditLog
Downloading audit logs for a DB instance	instance	ddsDownloadAuditLog
Deleting audit logs for a DB instance	instance	ddsDeleteAuditLogFile
Modifying recycling policy	instance	ddsModifyRecyclePolicy

15.2 Querying Traces

After you enable CTS, the system starts recording operations on cloud resources. CTS retains operation records generated in the last seven days.

This section describes how to check the operation records generated over the last seven days on the CTS console.

For details about how to view audit logs, see **Querying Real-Time Traces**.

15.3 Viewing Events

After CTS is enabled, the tracker starts recording operations on cloud resources. Operation records for the last 7 days are stored on the CTS console.

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

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

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

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

Step 5 Specify the filters used for querying traces. The following four filters are available:

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

  Select the filter from the drop-down list.

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

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

  When you select **Resource name** for **Search By**, you also need to select or enter a specific resource name.
- **Operator**: Select a specific operator (a user rather than tenant).
- **Trace Status**: Available options include **All trace statuses**, **normal**, **warning**, and **incident**. You can only select one of them.
- Start time and end time: You can specify the time period for query traces.

Step 6 Click ⌄ to the left of the record to be queried to extend its details.

Step 7 Locate a trace and click **View Trace** in the **Operation** column.

----End

16 DBA Assistant

16.1 Managing Sessions

Scenarios

You can manage sessions in the following scenarios:

- **Emergency Channel**: If the maximum number of connections for an instance has been reached and the instance cannot be logged in to, you can view and kill unnecessary sessions through this channel.
- **History Logs**: You can view history logs to learn details of the kill operations that you performed using the emergency channel.

Precautions

- This function is now in OBT. To use it, submit a service ticket.
- This function is not recommended unless you really need it. All your kill operations will be logged.
- DB instances of Community Edition 3.4, 4.0, 4.2, and 4.4 are supported.
- DB instances in the creating, frozen or abnormal state are not supported.
- Killing inactive sessions is not allowed.
- Real-time sessions are generated based on **currentOp** of a database at the current time point. If the execution time of a session is too short (less than or equal to milliseconds), you are not advised to view real-time sessions. If you need to collect statistics on all operations, see **Audit Log Policy Management**.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the cluster instance name.

Step 5 In the navigation tree, choose **DBA Assistant**.

Step 6 Choose **Real-Time Diagnosis**.

Step 7 Click **Real-Time Sessions**.

Step 8 On the displayed **Emergency Channel** page, view session statistics of the current instance node by summary, source, or namespace.

Step 9 By default, sessions are sorted and displayed in the session list in descending order by duration. You can also search for sessions by specifying **Sessions lasting longer than** or **Namespace**.

Step 10 Select the sessions that you want to kill and click **Kill Session**.

Step 11 In the **Kill Session** dialog box, confirm the session information and click **Yes**.

Step 12 Click **History Logs** to view the sessions killed through the emergency channel.

----End

17 Task Center

This section describes how to view the progress and result of asynchronous tasks on the **Task Center** page.

Precautions

Tasks that fail to be executed will be retained for seven days by default.

Tasks Overview

Table 17-1 List of tasks that can be viewed

Task Name	Description
Creating an instance	Creating a cluster instance or replica set instance.
Scaling up storage space	Scaling up the storage space of the shard node of a cluster instance or the storage space of a replica set instance.
Changing instance class	Changing the class of a cluster instance or replica set instance.
Adding nodes	Adding nodes to a cluster instance.
Restarting DB instances	Restarting a cluster instance, one or more cluster instance nodes, or a replica set instance.
Restoring to a new DB instance	Restoring data to a new cluster instance or replica set instance.
Restoring data to the original DB instance	Restoring data to a new Community Edition cluster instance, single node instance, or replica set instance.
Restoring to a point in time	Restoring a replica set instance to a point in time.

Task Name	Description
Restoring databases and tables to a point in time	Restores table-level data of a replica set instance to a specified point in time.
Performing a primary/standby switchover	Perform a primary/standby switchover for a replica set instance.
Binding and unbinding an EIP	Bind or unbind an EIP to or from a cluster instance, single node instance, or replica set instance.
Switching SSL	Enabling or disabling SSL for a cluster instance, single node instance, or replica set instance.
Changing a database port	Changing the database port of a cluster , single node, or replica set instance of Community Edition.
Changing a security group	Changing the security group of a cluster, single node, or replica set instance of Community Edition.
Changing a private IP address	Changing the private IP address of a cluster, single node, or replica set instance of Community Edition.
Changing an AZ	Changing the AZ of a cluster, single node, or replica set instance of Community Edition.
Enabling the shard/config IP address	Enabling the shard/config address for the cluster instance of Community Edition.
Modifying the oplog size	Changing the oplog size of a cluster, single node, or replica set instance of Community Edition
Physical backup	Creating automated and manual backups of a cluster, single node, or replica set instance of Community Edition
Upgrading minor version	Community Edition cluster and replica set instances are being patched.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 In the navigation pane on the left, click **Task Center**.

Step 5 In the navigation pane on the left, choose **Task Center**. Then, view the task progresses and results.

- You can view tasks in a specified period.
- The tasks can be located by DB instance name and ID or by task status or type from the drop-down list in the upper right corner.

----End

18 Billing

18.1 Renewing Instances

This section describes how to renew your yearly/monthly instances.

Precautions

- Pay-per-use DB instances do not involve renewals.
- Yearly/Monthly instances can only be renewed when their statuses are **Available**.

Renewing Instances

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, select the target instances and click **Renew** in the upper left corner of the instance list. In the displayed dialog box, click **Yes**.

Step 5 On the displayed page, renew the instances.

----End

Renewing an Instance

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the target instance and click **Renew** in the **Operation** column.

Step 5 On the displayed page, renew the instance.

----End

18.2 Changing the Billing Mode from Pay-per-Use to Yearly/Monthly

You can change the billing mode of an instance from pay-per-use to yearly/monthly to reduce your costs for using the instance for a long period of time.

Precautions

Only when the status of a pay-per-use instance is **Available**, its billing mode can be changed to yearly/monthly.

Changing Instance Billing in Batches

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, select the target instances and click **Change to Yearly/Monthly** above the instance list. In displayed dialog box, click **Yes**.

Step 5 On the displayed page, select how many months you wish to renew the instance for. The minimum duration is one month.

Confirm the settings and click **Pay**.

Step 6 Select a payment method and click **Pay**.

Step 7 View the results on the **Instances** page.

In the upper right corner of the instance list, click  to refresh the list. After the instance billing mode is changed to yearly/monthly, the instance status will change to **Available**. The billing mode becomes to **Yearly/Monthly**.

----End

Changing the Billing Mode of a Single Instance

Step 1 Log in to the management console.

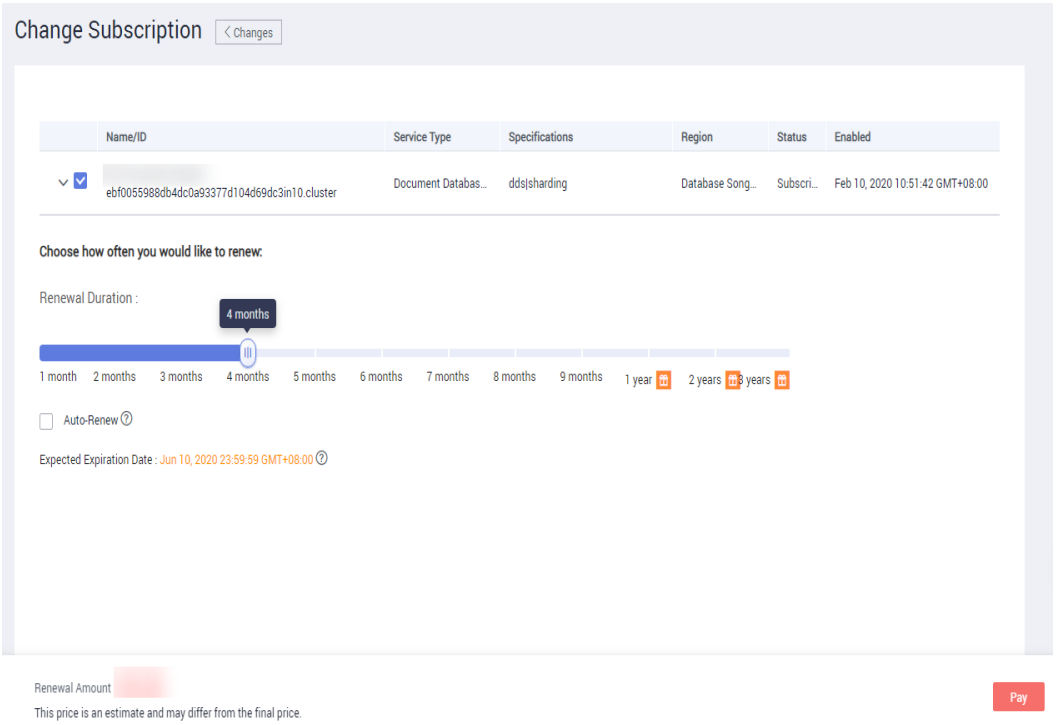
Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the target DB instance and in the **Operation** column, click **Change to Yearly/Monthly**.

Step 5 On the displayed page, select the renewal duration in month. The minimum duration is one month. **Figure 18-1** shows how to change a pay-per-use cluster instance to yearly/monthly.

Figure 18-1 Changing from pay-per-use to yearly/monthly



Confirm the settings and click **Pay**.

Step 6 Select a payment method and click **Pay**.

Step 7 View the results on the **Instances** page.

In the upper right corner of the instance list, click to refresh the list. After the instance billing mode is changed to yearly/monthly, the instance status will change to **Available**. The billing mode becomes to **Yearly/Monthly**.

----End

18.3 Changing the Billing Mode from Yearly/Monthly to Pay-per-Use

You can change yearly/monthly instances to pay-per-use instances on DDS and then pay only for the actual usage of your resources.

Precautions

- The billing mode can only be changed when the instance is in the **Available** status.

- Auto renewal will be disabled after the billing mode of your instances change to pay-per-use. Exercise caution when performing this operation.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, locate the target instance and click **Change to Pay-per-Use** in the **Operation** column.

Step 5 On the displayed page, confirm the instance information and click **Change to Pay-per-Use** to submit the change. The billing mode will change to pay-per-use after the instance expires.

Step 6 After you submit the change, a message is displayed in the **Billing Mode** column of the target instance, indicating that the billing mode will be changed to pay-per-use after the instance expires.

Step 7 To cancel the change, choose **Billing** > **Renewal** to enter the Billing Center. On the **Renewals** page, locate the target DB instance and click **More** > **Cancel Change to Pay-per-Use**.

Step 8 In the displayed dialog box, click **OK**.

----End

18.4 Unsubscribing from a Yearly/Monthly Instance

To unsubscribe from an instance billed on a yearly/monthly basis, you need to unsubscribe from the order.

Precautions

- If the DB instance is frozen, you can release the instance resource on the DDS console or in the billing center.
- To unsubscribe from an instance billed on a pay-per-use basis, you need to locate the instance and click **Delete** on the **Instances** page. For details, see **Deleting Pay-per-Use Instances**.
- Unsubscribe operations cannot be undone. Exercise caution when performing this operation. To retain data, create a manual backup before unsubscribing.

Method 1

Unsubscribe from a yearly/monthly instance on the **Instances** page.

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click ☰ in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, select instances and click **Unsubscribe** above the instance list. Alternatively, in the **Operation** column, choose **More** > **Unsubscribe**.

Step 5 In the displayed dialog box, click **Yes**.

Step 6 On the displayed page, confirm the order to be unsubscribed and select a reason. Then, click **Confirm**.

Step 7 In the displayed dialog box, click **Yes**.

NOTICE

1. After an unsubscription request is submitted, resources and data will be deleted and cannot be retrieved.
2. If you want to retain data, complete a manual backup before submitting the unsubscription request.

Step 8 View the results. After the DB instance order is successfully unsubscribed, the DB instance is no longer displayed in the instance list on the **Instances** page.

----End

Method 2

Unsubscribe from a yearly/monthly instance on the **Billing Center** page

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click ☰ in the upper left corner of the page and choose **Databases** > **Document Database Service**.

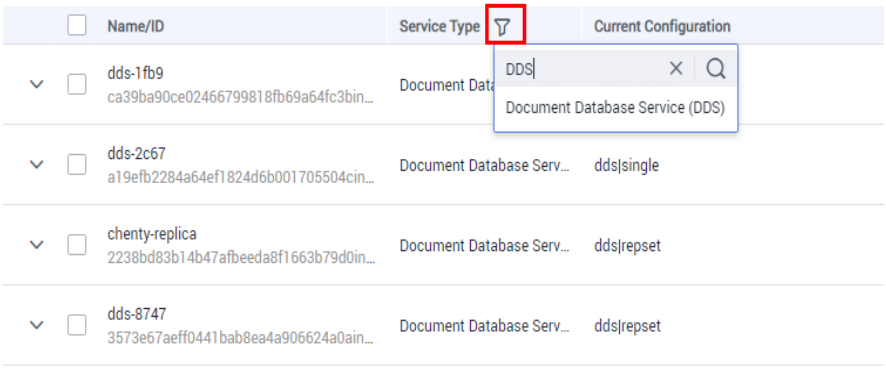
Step 4 In the upper right corner, click **Billing Center**.

Step 5 In the navigation pane on the left, choose **Orders** > **Unsubscriptions and Returns/Exchanges**.

Step 6 On the displayed **Cloud Service Unsubscriptions** page, select the order to be unsubscribed and click **Unsubscribe** in the **Operation** column.

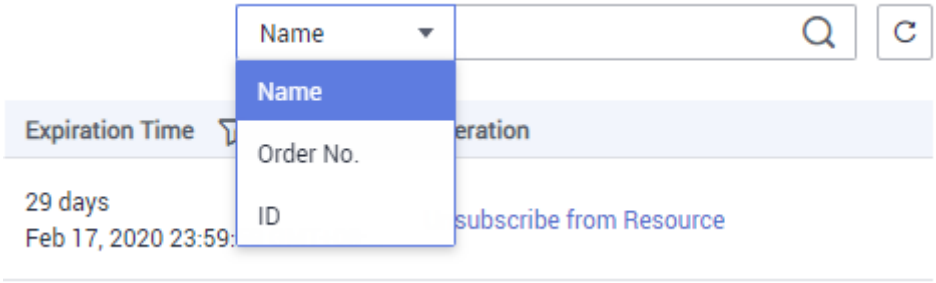
- You can select **Document Database Service (DDS)** in the **Service Type** to filter all DDS orders.

Figure 18-2 Filtering all orders



- Alternatively, you can search for orders by name, order No., or ID in the search box in the upper right corner of the order list.

Figure 18-3 Searching for orders



- A maximum of 20 orders can be unsubscribed from at a time.

Step 7 On the displayed page, confirm the order to be unsubscribed and select a reason. Then, click **Confirm**.

Step 8 In the displayed dialog box, click **Yes**.

NOTICE

1. After an unsubscription request is submitted, resources and data will be deleted and cannot be retrieved.
2. If you want to retain data, complete a manual backup before submitting the unsubscription request.

Step 9 View the results. After the DB instance order is successfully unsubscribed, the DB instance is no longer displayed in the instance list on the **Instances** page.

----End

19 DDS Tags

19.1 Adding or Modifying a Tag

Scenarios

Tags help you identify and manage DDS resources. When there are a large number of instances, you can add tags to them to quickly filter them. An instance can be tagged during or after it is created.

This section describes how to add and modify tags after an instance is created.

Precautions

- You are advised to set predefined tags on the TMS console.
- A tag consists of a key and value. You can add only one value for each key. For details about the naming rules of tag keys and tag values, see **Table 19-1**.
- Up to 20 tags can be added for a DB instance.
- Deleting tags of a DB instance has no adverse impact on the DB instance. After all tags of a DB instance are deleted, the DB instance cannot be filtered by tag.

Procedure

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 Click  in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the navigation pane on the left, click **Tags**.

Step 6 On the **Tags** page, click **Add Tag**. In the displayed dialog box, specify the tag key and value and click **OK**.

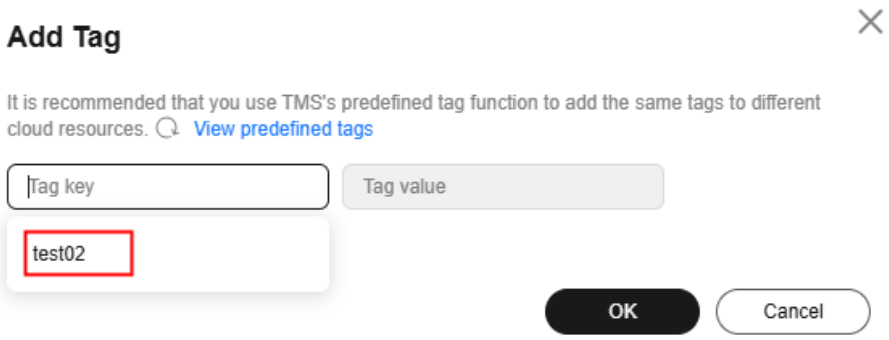
- Add a predefined tag.

  Predefined tags can be used to identify multiple cloud resources.

  To tag a cloud resource, you can select a created predefined tag from the drop-down list, without entering a key and value for the tag.

  For example, if a predefined tag has been created, its key is test02 and value is Project1. When you configure the key and value for a cloud resource, the created predefined tag will be automatically displayed on the page.

  Figure 19-1 Adding a predefined tag

- Create a tag.

  When creating a tag, enter the tag key and value.

  Figure 19-2 Adding a tag

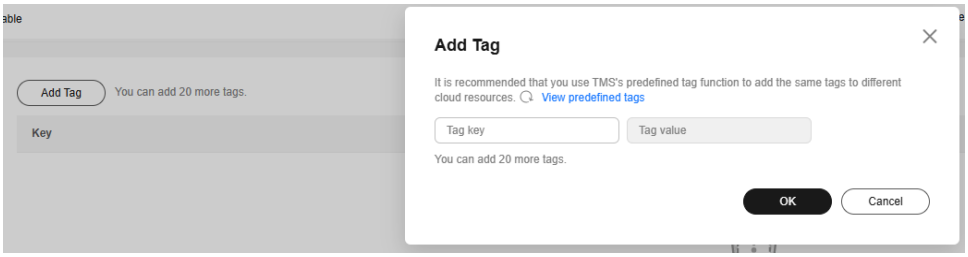
  

Table 19-1 Naming rules

Parameter	Requirement	Example
Tag key	– The key cannot be empty and contains 1 to 128 single-byte characters. – The key can contain UTF-8 letters (including Chinese characters), digits, spaces, and the following characters: _.:/=+-@ – The key cannot start with **sys**, and cannot start or end with a space. – The key can only consist of digits, letters, underscores (_), and hyphens (-).	Organization
Tag value	– The value can contain UTF-8 letters (including Chinese characters), digits, spaces, and the following characters: _.:/=+-@ – The value can be empty or null and contains 0 to 255 single-byte characters. – The value can only consist of digits, letters, underscores (_), periods (.), and hyphens (-).	dds_01

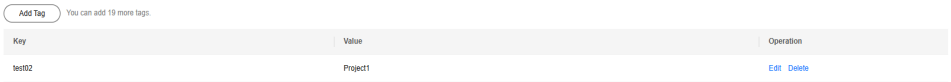
Step 7 View and manage tags on the **Tags** page.

You can click **Edit** in the **Operation** column to change the tag value.

NOTE

Only the tag value can be edited when editing a tag.

Figure 19-3 Tag added



----End

19.2 Filtering Instances by Tag

Scenarios

After tags are added, you can filter instances by tag to quickly find instances of a specified category.

Procedure

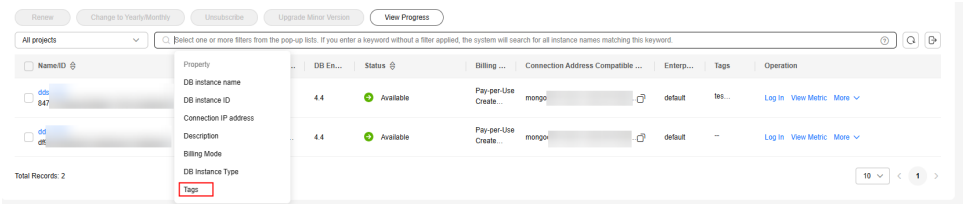
Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

Step 3 Click in the upper left corner of the page and choose **Databases** > **Document Database Service**.

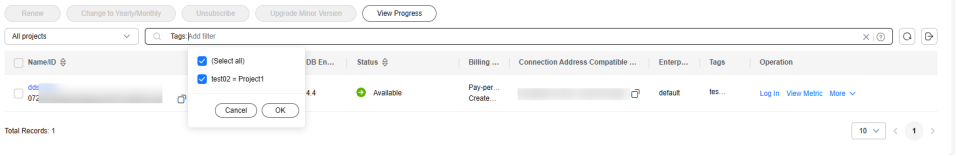
Step 4 On the **Instances** page, click the search box above the instance list and select **Tags**.

Figure 19-4 Searching by tag



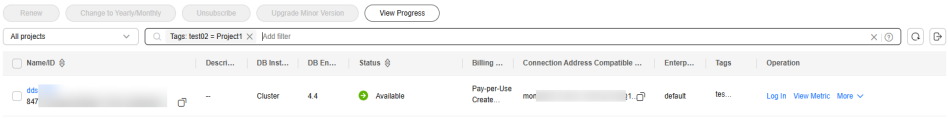
Step 5 Enter the tag key and value associated with the instance and click **OK**.

Figure 19-5 Entering the tag key and value



Step 6 View the instance information.

Figure 19-6 Viewing instance information



----End

19.3 Deleting a Tag

Scenarios

If a tag is no longer needed, you can delete the tag to unbind it from the instance.

Procedure

Step 1 Log in to the management console.

Step 2 Click in the upper left corner and select a region and a project.

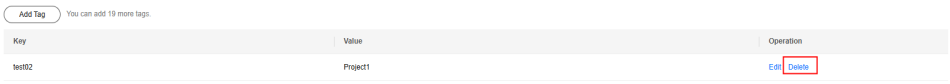
Step 3 Click ☰ in the upper left corner of the page and choose **Databases** > **Document Database Service**.

Step 4 On the **Instances** page, click the instance name.

Step 5 In the navigation pane on the left, click **Tags**.

Step 6 On the **Tags** page, locate the tag to be deleted and click **Delete** in the **Operation** column. In the displayed dialog box, click **Yes**.

Figure 19-7 Deleting a tag



Step 7 After the tag is deleted, it is no longer displayed on the **Tags** page.

----End

20 DDS Quotas

Scenarios

A quota is a limit on the quantity or capacity of a certain type of service resources available to you. For example, the maximum number of DDS DB instances that can be created varies depending on the DB instance type. If a quota cannot meet your needs, apply for a higher quota.

This section describes how to view the usage of each type of DDS resource and the total quotas in a specified region.

Viewing Quotas

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 In the upper right corner of the DDS console, choose **Resources** > **My Quota**.

Step 4 View the used and total quota of each type of DDS resource.

----End

Increasing Quotas

Step 1 Log in to the management console.

Step 2 Click  in the upper left corner and select a region and a project.

Step 3 In the upper right corner of the DDS console, choose **Resources** > **My Quota**.

Step 4 Click **Increase Quota**.

Step 5 On the **Create Service Ticket** page, configure parameters as required.

In the **Problem Description** area, fill in the content and reason for adjustment.

Step 6 After all required parameters are configured, select the agreement and click **Submit**.

----End

21 DDS Usage Suggestions

21.1 Design Rules

Naming

- The name of a database object (database name, table name, field name, or index name) has to start with a lowercase letter and must be followed by a letter or digit. The length of the name cannot exceed 32 bytes.
- The database name cannot contain special characters (""$\/*?~#:|") or null character (\0). The database name cannot be a system database name, such as **admin**, **local**, and **config**.
- The database collection name can only contain letters and underscores (_). The name cannot be prefixed with "system". The total length of <*Database name*>.<*Collection name*> cannot exceed 120 characters.

Index

You can use indexes to avoid full table scans and improve query performance.

- A column index can have up to 512 bytes, an index name can have up to 64 characters, and a composite index can have up to 16 columns.
- The total length of <*Database name*>.<*Collection name*>.$<*Index name*> cannot exceed 128 characters.
- Create indexes for fields with high selectivity. If you create indexes for low selective fields, large result sets may be returned. This should be avoided.
- Write operations on a collection will trigger more I/O operations on indexes in the collection. Ensure that the number of indexes in a collection does not exceed 32.
- Do not create indexes that will not be used. Unused indexes loaded to the memory will cause a waste of memory. In addition, useless indexes generated due to changes in service logic must be deleted in a timely manner.
- Indexes must be created in the background instead of foreground. When creating an index in the foreground of a collection, you continuously hold the exclusive lock of the parent database. This will block all read and write operations on the database and all its collections until the index is created. To

avoid such blocking, you are advised to use the **background: true** option to create an index. For details, see **Creating and Managing Indexes**.

- An index must be created for the sort key. If a composite index is created, the column sequence of the index must be the same as that of the sort key. Otherwise, the index will not be used.
- Do not create an index based on the leading-edge column of a composite index. If the leading-edge column of a composite index is the column used in another index, the smaller index can be removed. For example, a composite index based on "firstname" and "lastname" can be used for queries on "firstname". In this case, creating another firstname-based index is unnecessary.
- Creating indexes consumes a lot of I/O and compute resources. You are advised to create indexes during off-peak hours. Do not concurrently create more than five indexes. If you need to create multiple indexes for a given collection, run the **createIndexes** command to deliver multiple indexes at a time to reduce performance loss.
- Before deleting an index, check whether the index is being created on the secondary node. If the index is being created on the secondary node, do not delete the index immediately. Otherwise, the secondary node may be in lock waiting, causing unexpected problems.

Sharding

You can shard collections to maximize the cluster performance.

Suggestions for sharding collections:

- In scenarios where the data volume is large (more than one million rows) and the write/read ratio is high, sharding is recommended if the data volume increases with the service volume.
- If you shard a collection using a hashed shard key, pre-splitting the chunks of the sharded collection can help reduce the impact of automatic balancing and splitting on service running.
- If sharding is enabled for a non-empty collection, the collection is locked and services are blocked. The time required for sharding increases with the amount of data in the collection. To minimize the impact, before sharding, ensure that the target table's workload is running during off-peak hours.
- If sharding is enabled for a non-empty collection, the time window for enabling the balancer must be set during off-peak hours. Otherwise conflicts may occur during data balancing between shards and service performance will be affected.
- If you want to perform a sort query based on the shard key and new data is evenly distributed based on the shard key, you can use ranged sharding. In other scenarios, you can use hashed sharding.
- Properly design shard keys to prevent a large amount of data from using the same shard key, which may lead to jumbo chunks.
- If a sharded cluster is used, you must run **flushRouterConfig** after running **dropDatabase**. For details, see **How Do I Prevent dds mongos Cache Problem?**
- After existing data is sharded, if the **filter** field of the update request does not contain shard keys and **upsert:true** or **multi:false**, the update request will

report an error and return message "An upsert on a sharded collection must contain the shard key and have the simple collation."

21.2 Development Rules

Database Connections

If the maximum number of mongod or dds mongos connections is reached, your client cannot connect to the DDS instances. Each connection received by mongod or dds mongos is processed by a single thread of 1 MB stack space. As the connections increase, too many threads will increase the context switching overhead and memory usage.

- If you connect to databases from clients, calculate the number of clients and the size of the connection pool configured for each client. The total number of connections cannot exceed 80% of the maximum number of connections allowed by the current instance.
- The connection between the client and the database must be stable. It is recommended that the number of new connections per second be less than 10.
- You are advised to set the connection timeout interval of the client to at least three times the maximum service execution duration.
- For a replica set instance, the IP addresses of both the primary and standby nodes must be configured on the client. For a cluster instance, at least two dds mongos IP addresses must be configured.
- DDS uses user **rwuser** by default. When you log in as user **rwuser**, the authentication database must be **admin**.

Reliability

Rules for setting write concern: For mission-critical services, set write concern to {w:n},n>0. A larger value is better consistency but poorer performance.

- **w:1** means that a confirmation message was returned after data was written to the primary node.
- **w:1,journal:true** means that the result was returned after data was written to the primary node and logs.
- **w:majority** means that the result was returned after data was written to more than half of the total standby nodes.

  NOTE

  If data is not written using **w:majority**, the data that is not synchronized to the standby node may be lost when a primary/standby switchover occurs.

If high reliability is required, deploy a cluster in three AZs.

Performance

Specification

- The service program is not allowed to perform full table scanning.

- During the query, select only the fields that need to be returned. In this way, the network and thread processing loads are reduced. If you need to modify data, modify only the fields that need to be modified. Do not directly modify the entire object.
- Do not use $not. DDS does not index missing data. The $not query requires that all records be scanned in a single result collection. If $not is the only query condition, a full table scan will be performed on the collection.
- If you use $and, put the conditions with the fewest matches before other conditions. If you use $or, put the conditions with the more matches first.
- In a DB instance, the total number of databases cannot exceed 200, and the total number of collections cannot exceed 500. If the number of collections is too large, the memory may be overloaded. In addition, the performance for restarting a DB instance and performing a primary/standby switchover may deteriorate due to too many collections, which affects the high availability performance in emergencies.
- Before bringing a service online, perform a load test to measure the performance of the database in peak hours.
- Do not execute a large number of concurrent transactions at the same time or leave a transaction uncommitted for a long time.
- Before rolling out services, execute query plans to check the query performance for all query types.

Suggestions

- Each connection is processed by an independent thread in the background. Each thread is allocated with 1 MB stack memory. The number of connections should not be too large. Otherwise, too much memory is occupied.
- Use the connection pool to avoid frequent connection and disconnection. Otherwise, the CPU usage is too high.
- Reduce disk read and write operations: Reduce unnecessary upsert operations.
- Optimize data distribution: Data is sharded and hot data is distributed evenly between shards.
- Reduce lock conflicts: Do not perform operations on the same key too frequently.
- Reduce lock wait time: Do not create indexes on the frontend.

Notice

During the development process, each execution on a collection must be checked using explain() to view its execution plan. Example:

db.T_DeviceData.find({"deviceId":"ae4b5769-896f"}).explain();

db.T_DeviceData.find({"deviceId":"77557c2-31b4"}).explain("executionStats");

A covered query does not have to read a document and returns a result from an index, so using a covered query can greatly improve query efficiency. If the output of explain() shows that indexOnly is true, the query is covered by an index.

Execution plan parsing:

1. Check the execution time. The smaller the values of the following parameters, the better the performance:

executionStats.executionStages.executionTimeMillisEstimate and **executionStats.executionStages.inputStage. executionTimeMillisEstimate**

- **executionStats.executionTimeMillis** specifies how much time the database took to both select and execute the winning plan.
- **executionStats.executionStages.executionTimeMillisEstimate** specifies the execution completion time of the execution plan.
- **executionStats.executionStages.inputStage. executionTimeMillisEstimate** specifies the execution completion time of the sub-phase of the execution plan.

2. Check the number of scanned records. If the three items are the same, the index is best used.
   - **executionStats. nReturned** is the number of documents that match the query condition.
   - **executionStats .totalKeysExamined** indicates the number of scanned index entries.
   - **executionStats .totalDocsExamined** indicates the number of scanned document entries.
3. Check the stage status. The following combinations of stages can provide good performance.
   - Fetch+IDHACK
   - Fetch+ixscan
   - Limit+ (Fetch+ixscan)
   - PROJECTION+ixscan

Table 21-1 Status description

Status Name	Description
COLLSCAN	Full table scan
SORT	In-memory sorting
IDHACK	_id-based query
TEXT	Full-text index
COUNTSCAN	Number of unused indexes
FETCH	Index scanning
LIMIT	Using Limit to limit the number of returned records
SUBPLA	$or query stage without using an index
PROJECTION	Restricting the return of stage when a field is returned.
COUNT_SCAN	Number of used indexes

Cursor Usage Rules

If a cursor is inactive for 10 minutes, it will be automatically closed. You can also manually close it to save resources.

Rules for Using Distributed Transactions in Version 4.2

- Spring Data MongoDB does not support the retry mechanism after a transaction error is reported. If the client uses Spring Data MongoDB as the client to connect to MongoDB, you need to use Spring Retry to retry the transaction based on the references of Spring Data MongoDB.
- The size of the distributed transaction operation data cannot exceed 16 MB.

Precautions for Backups

Do not perform DDL operations during the backup to avoid backup failures.

