

# Alibaba Cloud

## ApsaraDB for MongoDB Quick Start for Cluster









Document Version: 20201103

# Legal disclaimer

Alibaba Cloud reminds you to carefully read and fully understand the terms and conditions of this legal disclaimer before you read or use this document. If you have read or used this document, it shall be deemed as your total acceptance of this legal disclaimer.

1. You shall download and obtain this document from the Alibaba Cloud website or other Alibaba Cloud-authorized channels, and use this document for your own legal business activities only. The content of this document is considered confidential information of Alibaba Cloud. You shall strictly abide by the confidentiality obligations. No part of this document shall be disclosed or provided to any third party for use without the prior written consent of Alibaba Cloud.
2. No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company or individual in any form or by any means without the prior written consent of Alibaba Cloud.
3. The content of this document may be changed because of product version upgrade, adjustment, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and an updated version of this document will be released through Alibaba Cloud-authorized channels from time to time. You should pay attention to the version changes of this document as they occur and download and obtain the most up-to-date version of this document from Alibaba Cloud-authorized channels.
4. This document serves only as a reference guide for your use of Alibaba Cloud products and services. Alibaba Cloud provides this document based on the "status quo", "being defective", and "existing functions" of its products and services. Alibaba Cloud makes every effort to provide relevant operational guidance based on existing technologies. However, Alibaba Cloud hereby makes a clear statement that it in no way guarantees the accuracy, integrity, applicability, and reliability of the content of this document, either explicitly or implicitly. Alibaba Cloud shall not take legal responsibility for any errors or lost profits incurred by any organization, company, or individual arising from download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, take responsibility for any indirect, consequential, punitive, contingent, special, or punitive damages, including lost profits arising from the use or trust in this document (even if Alibaba Cloud has been notified of the possibility of such a loss).
5. By law, all the contents in Alibaba Cloud documents, including but not limited to pictures, architecture design, page layout, and text description, are intellectual property of Alibaba Cloud and/or its affiliates. This intellectual property includes, but is not limited to, trademark rights, patent rights, copyrights, and trade secrets. No part of this document shall be used, modified, reproduced, publicly transmitted, changed, disseminated, distributed, or published without the prior written consent of Alibaba Cloud and/or its affiliates. The names owned by Alibaba Cloud shall not be used, published, or reproduced for marketing, advertising, promotion, or other purposes without the prior written consent of Alibaba Cloud. The names owned by Alibaba Cloud include, but are not limited to, "Alibaba Cloud", "Aliyun", "HiChina", and other brands of Alibaba Cloud and/or its affiliates, which appear separately or in combination, as well as the auxiliary signs and patterns of the preceding brands, or anything similar to the company names, trade names, trademarks, product or service names, domain names, patterns, logos, marks, signs, or special descriptions that third parties identify as Alibaba Cloud and/or its affiliates.
6. Please directly contact Alibaba Cloud for any errors of this document.

# Document conventions

| Style  | Description   | Example   |
|--|---|---|
|  <b>Danger</b>  | A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results. |  <b>Danger:</b><br>Resetting will result in the loss of user configuration data.                                       |
|  <b>Warning</b> | A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results. |  <b>Warning:</b><br>Restarting will cause business interruption. About 10 minutes are required to restart an instance. |
|  <b>Notice</b>  | A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.      |  <b>Notice:</b><br>If the weight is set to 0, the server no longer receives new requests.                              |
|  <b>Note</b>  | A note indicates supplemental instructions, best practices, tips, and other content.  |  <b>Note:</b><br>You can use Ctrl + A to select all files.  |
| >  | Closing angle brackets are used to indicate a multi-level menu cascade.   | Click <b>Settings</b> > <b>Network</b> > <b>Set network type</b> .  |
| <b>Bold</b>  | Bold formatting is used for buttons, menus, page names, and other UI elements.  | Click <b>OK</b> .   |
| <code>Courier font</code>  | Courier font is used for commands   | Run the <code>cd /d C:/window</code> command to enter the Windows system folder.  |
| <i>Italic</i>  | Italic formatting is used for parameters and variables.   | <code>bae log list --instanceid</code><br><i>Instance_ID</i>  |
| [ ] or [a b]   | This format is used for an optional value, where only one item can be selected.   | <code>ipconfig [-all -t]</code>   |
| { } or {a b}   | This format is used for a required value, where only one item can be selected.  | <code>switch {active stand}</code>  |

# Table of Contents

|   |    |
|---|----|
| 1. Notes .....  | 05 |
| 2. ApsaraDB for MongoDB console .....                               | 06 |
| 3. Get started with a sharded cluster instance .....                | 08 |
| 4. Create a sharded cluster instance .....                          | 09 |
| 5. Set a password for a sharded cluster instance .....              | 16 |
| 6. Configure a whitelist for a sharded cluster instance .....       | 17 |
| 7. Apply for a public endpoint for a sharded cluster instance ..... | 19 |
| 8. Connect to an instance .....                                     | 22 |
| 8.1. Overview of sharded cluster instance connections .....         | 22 |
| 8.2. Connect to a sharded cluster ApsaraDB for MongoDB insta.....   | 24 |
| 8.3. Connect to a sharded cluster instance by using the mong.....   | 25 |
| 8.4. Connect to an ApsaraDB for MongoDB instance through t.....     | 27 |
| 9. Migrate data .....   | 34 |
| 9.1. Migrate a user-created sharded MongoDB database to Aps.....    | 34 |
| 9.2. Migrate a user-created MongoDB database to ApsaraDB fo.....    | 40 |

# 1. Notes

You can migrate data from a user-created MongoDB database to an ApsaraDB for MongoDB instance. Notice the limits of ApsaraDB for MongoDB.

| Operation                                       | Limit   |
|---|---|
| Deploy a sharded cluster instance               | The database version must match the storage engine. For more information, see <a href="#">MongoDB versions and storage engines</a> .  |
| Build cluster components                        | <ul style="list-style-type: none"> <li>When you create a sharded cluster instance, you can specify the specifications and numbers of mongos and shards.</li> <li>While the instance is running, you can add mongos and shards, but you cannot remove them. For more information, see <a href="#">Configuration change overview</a>.</li> </ul>  |
| Restart a sharded cluster instance              | You must restart the instance in the <a href="#">ApsaraDB for MongoDB console</a> or by calling the <a href="#">RestartDBInstance</a> operation.  |
| Migrate data from a sharded cluster instance    | You can use the built-in commands of MongoDB or Data Transmission Service (DTS) to migrate data. For more information, see <a href="#">Use the built-in commands of MongoDB to migrate data of a sharded cluster instance</a> or <a href="#">Use DTS to migrate data of a sharded cluster instance</a> .  |
| Back up the data of a sharded cluster instance  | <p><a href="#">Configure automatic backup for an ApsaraDB for MongoDB instance</a>: Only physical backup is supported. <a href="#">Manually back up an ApsaraDB for MongoDB instance</a>: Both physical backup and logical backup are supported.</p> <div style="background-color: #e6f2ff; padding: 10px; border: 1px solid #d9e1f2;"> <p> <b>Note</b> If the database version of the instance is 3.2 or 3.4, the number of collections and indexes in the instance cannot exceed 10,000. Otherwise, physical backup may fail. If you want to increase this limit, we recommend that you upgrade the database version to 4.0. For more information, see <a href="#">Upgrade the database version of an ApsaraDB for MongoDB instance</a>. Alternatively, you can select the database version 4.0 when you create the instance.</p> </div> |
| Restore data to a sharded cluster instance      | You can only restore data by point in time. For more information, see <a href="#">Restore data to a new ApsaraDB for MongoDB instance by point in time</a> .  |
| Modify parameters of a sharded cluster instance | For security and stability, you are not allowed to modify the parameters of a sharded cluster instance.   |
| Data read and write                             | You can only read data from the admin database of the sharded cluster instance, but you cannot write data to the admin database.  |

## 2.ApsaraDB for MongoDB console

The ApsaraDB for MongoDB console is a web application for managing instances. In the ApsaraDB for MongoDB console, you can create and manage instances, configure the instance IP whitelists, passwords, and network types, and perform other operations.

The ApsaraDB for MongoDB console is part of the Alibaba Cloud Management Console. For more information about common settings and basic operations in the Alibaba Cloud Management Console, see [Alibaba Cloud Management Console](#).

### Prerequisites

An Alibaba Cloud account is used. To create an Alibaba Cloud account, go to the [Alibaba Cloud official website](#).

### ApsaraDB for MongoDB console homepage

The console homepage displays the same information for all sharded cluster instances.

Log on to the [ApsaraDB for MongoDB console](#) and go to the **Sharded Cluster Instances** page, as shown in the following figure. This figure is only to be used for reference. The actual page may be different.



UI element description

| No. | UI element                | Description  |
|-----|---------------------------|--|
| 1   | Sharded Cluster Instances | The ApsaraDB for MongoDB console homepage, which displays all sharded cluster instances in a region that belong to the current account.  |
| 2   | Region                    | You can click a region to display all instances that reside within the region.   |
| 3   | Refresh                   | The button to refresh the instance information page.   |
| 4   | Create Instance           | The button to <a href="#">create an instance</a> .   |
| 5   | Instance ID               | <ul style="list-style-type: none"> <li>You can click an instance ID to go to the Basic Information page of the instance.</li> <li>You can click the icon following an instance ID to modify the name of the instance.</li> </ul> |
| 6   | Status                    | The status of the instance. Instances may be in different states.  |
| 7   | More icon                 | You can click this icon to manage, restart, or release an instance.  |
| 8   | Export                    | The button to <a href="#">Export the list of instances</a> .   |

### ApsaraDB for MongoDB instance console

Log on to the [ApsaraDB for MongoDB console](#). Click an **Instance ID** or **Manage** in the Actions column corresponding to an instance. The **Basic Information** page is displayed. The following table lists the UI elements on the page.

| UI element or page  | Section                   | Description  | Operation  |
|---------------------|---------------------------|--|--|
| Top navigation bar  | N/A                       | You can back up and restart the instance.  | <ul style="list-style-type: none"> <li>• <a href="#">Back up an instance</a></li> <li>• <a href="#">Restart an instance</a></li> </ul>   |
| Basic Information   | Basic Information         | You can view the basic information about the instance, such as the instance ID, region, network type, and storage engine.  | N/A  |
|                     | Specification Information | You can view the instance specifications such as the database version, maintenance period, billing method, creation time, and expiration time.   | <a href="#">Specify a maintenance period</a>   |
|                     | Mongos List or Shard List | <ul style="list-style-type: none"> <li>• In the mongos list, you can find a mongos ID and click the <ul style="list-style-type: none"> <li>□ icon to change its configurations, and log on to or restart the mongos.</li> </ul> </li> <li>• In the shard list, you can find a shard ID and click the <ul style="list-style-type: none"> <li>□ icon to log on to a shard, restart the shard, trigger a failover, or change the shard configurations.</li> </ul> </li> <li>• Database read and write operations may fail when you restart nodes. We recommend that you do not perform the Create, Retrieve, Update, and Delete (CRUD) operations on databases when you restart nodes.</li> </ul> | <ul style="list-style-type: none"> <li>• <a href="#">Trigger a failover</a></li> <li>• <a href="#">Change Specifications</a></li> <li>• <a href="#">Log on to a database</a></li> <li>• <a href="#">Restart a node</a></li> <li>• <a href="#">View monitoring information</a></li> </ul> |
| Backup and Recovery | N/A                       | You can view and download a list of data backups for a specified time period, restore data from the specified time period, or create an instance from a specified backup point.  | <ul style="list-style-type: none"> <li>• <a href="#">Download backup data</a></li> <li>• <a href="#">Restore data to a new ApsaraDB for MongoDB instance by point in time</a></li> </ul>   |
| Monitoring Info     | N/A                       | You can view monitoring information of mongos or shards based on the specified metrics and time range.   | N/A  |
| Data Security       | Whitelist Settings        | You can configure an IP whitelist.   | <a href="#">Configure a whitelist</a>  |
|                     | Audit Logs                | Audit logs record all operations that you perform on a database. You can use these logs in analysis.   | <a href="#">View audit logs</a>  |

## 3. Get started with a sharded cluster instance

This topic describes how to get started with a sharded cluster instance of ApsaraDB for MongoDB. In this topic, you can quickly become familiar with the operations for creating, configuring, and connecting to an ApsaraDB for MongoDB instance.

### Flowchart for an ApsaraDB for MongoDB instance

The first time you use ApsaraDB for MongoDB, you can start with [Notes](#).

The following figure shows the operations that you must perform before you use an ApsaraDB for MongoDB instance.



1. [Create a sharded cluster instance](#).
2. [Set a password for a sharded cluster instance](#).
3. [Configure a whitelist for a sharded cluster instance](#).
4. (Optional) [Apply for a public endpoint for a sharded cluster instance](#).
5. Connect to a sharded cluster instance. For more information, see [Overview of sharded cluster instance connections](#).



## 4. Create a sharded cluster instance

This topic describes how to create a sharded cluster instance in the ApsaraDB for MongoDB console.

### Prerequisites

- An Alibaba Cloud account is registered. For more information, see [Sign up with Alibaba Cloud](#).
- Your account balance is sufficient if you want to create a pay-as-you-go instance.

### Billing

For more information, see [Billing items and pricing](#).

### Procedure

1. Log on to the [ApsaraDB for MongoDB console](#).
2. In the upper-left corner of the page, select the resource group and the region of the target instance.
3. In the left-side navigation pane, click **Sharded Cluster Instances**.
4. On the **Sharded Cluster Instances** page, click **Create Instance**.
5. Click **Subscription(Sharding)** or **Pay-As-You-Go(Sharding)**.



#### Note

- **Subscription:** You must pay the subscription fee when you create an instance. We recommend that you select this billing method for long-term use, because it is more cost-effective than pay-as-you-go billing. Longer subscription periods have larger discounts.
- **Pay-as-you-go:** A pay-as-you-go instance is charged at an hourly rate based on your actual resource usage. We recommend that you select this billing method for short-term use. You can reduce costs by releasing your pay-as-you-go instance after you no longer need it.

6. Configure the instance parameters. The following table describes related parameters.

### Instance parameters

| Section | Parameter     | Description   |
|---------|---------------|---|
|         | <b>Region</b> | <p>The region where the sharded cluster instance is deployed. After an instance is created, you cannot change its region. Exercise caution when you select the region.</p> <p>Only instances in the same region (for example, an <a href="#">ECS</a> and an ApsaraDB for MongoDB instance) can be interconnected with each other over the internal network.</p> |

| Section | Parameter        | Description   |
|---------|------------------|---|
|         | Zone             | <p>A <b>zone</b> is a geographic area with an independent power supply and network. An ECS instance and an ApsaraDB for MongoDB instance in the same zone can be interconnected over the internal network with the minimum network latency.</p> <div data-bbox="683 450 1385 624" style="background-color: #e1f5fe; padding: 5px;"> <p> <b>Note</b> To implement zone-disaster recovery, you can deploy the replica set instance across multiple zones. For more information, see <a href="#">Create a multi-zone sharded cluster instance</a>.</p> </div> |
|         | Database Version | <p>The version of the database engine for the sharded cluster instance. ApsaraDB for MongoDB supports 3.2, 3.4, 4.0, and 4.2.</p> <p>For more information about versions, see <a href="#">MongoDB versions and storage engines</a>.</p> <div data-bbox="683 860 1385 1003" style="background-color: #e1f5fe; padding: 5px;"> <p> <b>Note</b> You can manually upgrade the database version while an instance is running. For more information, see <a href="#">Upgrade the database version</a>.</p> </div>  |
|         | Storage Engine   | <p>The storage engine is <b>WiredTiger</b>.</p>   |
|         |                  |   |

| Section                           | Parameter                  | Description           |
|-----------------------------------|----------------------------|-----------------------|
| <p><b>Basic Configuration</b></p> | <p><b>Network Type</b></p> | <p><b>Classic</b></p> |

| Section | Parameter | Description |
|---------|-----------|-------------|
|         |           |             |

| Section              | Parameter     | Description   |
|----------------------|---------------|---|
| VPC                  |               | <p>A VPC is an isolated network with higher security and performance than the classic network. We recommend that you select the VPC network type.</p> <div style="background-color: #e1f5fe; padding: 10px; border: 1px solid #cfcfcf;"> <p><b>Note</b></p> <ul style="list-style-type: none"> <li>◦ You must create a VPC in advance. For more information, see <a href="#">Create a VPC</a>.</li> <li>◦ You can change the network type after you create an instance. For more information, see <a href="#">Modify the instance network type</a>.</li> <li>◦ If you want to migrate your applications to the cloud, you can build a virtual data center by connecting your on-premises data center to the resources in a VPC through a leased line or a virtual private network (VPN). For more information, see <a href="#">Configure a hybrid access solution to switch the network type of an ApsaraDB for MongoDB instance from Classic Network to VPC</a>.</li> </ul> </div> |
| Mongos Specification | Specification | <p>The specifications of each mongos. For more information, see <a href="#">Instance specifications</a>.</p> <div style="background-color: #e1f5fe; padding: 10px; border: 1px solid #cfcfcf;"> <p><b>Note</b> You can add new mongos or change the configurations of existing mongos while a sharded cluster instance is running.</p> </div>   |
|                      | Quantity      | <p>The number of mongos.</p> <p>A sharded cluster instance can contain 2 to 32 mongos.</p>  |
| Shard Specifications | Specification | <p>The specifications of each shard. For more information, see <a href="#">Instance types</a>.</p> <div style="background-color: #e1f5fe; padding: 10px; border: 1px solid #cfcfcf;"> <p><b>Note</b> You can add new shards or change the configurations of existing shards while a sharded cluster instance is running.</p> </div>   |
|                      | Storage Space | <p>The storage space of each shard. Valid values: 10 GB to 1,000 GB.</p> <div style="background-color: #e1f5fe; padding: 10px; border: 1px solid #cfcfcf;"> <p><b>Note</b> The storage space of a shard stores your data, system, and log files.</p> </div>   |

| Section                            | Parameter            | Description  |
|------------------------------------|----------------------|--|
|                                    | <b>Quantity</b>      | The number of shards.<br>A sharded cluster instance can contain 2 to 32 shards.  |
| <b>Configserver Specifications</b> | <b>Specification</b> | The specifications of each config server: 1 core, 2 GB of memory, and 20 GB of storage space. The config server specifications are not user-configurable.  |
| <b>Set Password</b>                | <b>Set Now</b>       | <ul style="list-style-type: none"> <li>The password must contain at least three of the following character types: uppercase letters, lowercase letters, digits, and special characters. Special characters include <code>!#\$%^&amp;*()_+ -=</code></li> <li>The password must be 8 to 32 characters in length.</li> </ul> |
|                                    | <b>Set Later</b>     | <a href="#">Set a password for a sharded cluster instance</a> after the instance is created.   |
| <b>Validity</b>                    | N/A                  | You must specify this parameter when you create a sharded cluster instance that uses the subscription billing method. You can select one to nine months for a monthly subscription or one to three years for an annual subscription.   |

- After you configure the preceding parameters, click **Buy Now**.
- On the **Confirm Order** page, read and select **ApsaraDB for MongoDB Agreement of Service**, and complete the payment.

## View the created instance

- Log on to the [ApsaraDB for MongoDB console](#).
- In the upper-left corner of the page, select the resource group and the region of the target instance.
- In the left-side navigation pane, click **Sharded Cluster Instances**.

## Troubleshoot if you cannot find the instance

| Possible causes                               | Solution   |
|---|--|
| You selected the wrong region in the console. | Select the region where the instance is deployed. For more information, see <a href="#">View the created instance</a> .                          |
| You opened the incorrect page.                | In the left-side navigation pane, click <b>Sharded Cluster Instances</b> . For more information, see <a href="#">View the created instance</a> . |

| Possible causes   | Solution  |
|---|---|
| The instance list in the ApsaraDB for MongoDB console was not updated or was updated before the instance was created. | Wait for several minutes and then update the instance list to check whether the instance is added to the list.  |
| Resources are insufficient.   | The system may fail to create the instance due to insufficient resources. In this case, your payment is refunded. You can check the refund on the <a href="#">Orders</a> page.<br>After you confirm the refunded fees, you can try to create the instance in another zone. You can also <a href="#">submit a ticket</a> . |

## What's next

After you create an instance, you must configure a whitelist. For more information, see [Configure a whitelist for a sharded cluster instance](#). If you want to connect to the instance over the Internet, you must apply for a public endpoint. For more information, see [Apply for a public endpoint for a sharded cluster instance](#).

For more information about instance connection methods and connection scenarios, see [Connect to an ApsaraDB for MongoDB instance](#).

# 5. Set a password for a sharded cluster instance

This topic describes how to set or reset a password for a sharded cluster instance.

## Procedure

1. Log on to the [ApsaraDB for MongoDB console](#).
2. In the upper-left corner of the page, select the resource group and the region of the target instance.
3. In the left-side navigation pane, click **Sharded Cluster Instances**.
4. Find the target instance and click its ID.
5. In the left-side navigation pane, click **Accounts**.
6. Click **Reset Password**.

7. In the **Reset Password** dialog box that appears, enter and confirm the new password. Click **OK**.

### Note

- The password must contain at least three types of the following characters: uppercase letters, lowercase letters, digits, and special characters. Special characters include ! # \$ % ^ & \* ( ) \_ + - =
- The password must be 8 to 32 characters in length.



## 6. Configure a whitelist for a sharded cluster instance

This topic describes how to configure a whitelist for a sharded cluster instance after you create the instance. Only the devices whose IP addresses are added to the whitelists of the instance are allowed access to the instance. The default whitelist only contains the IP address 127.0.0.1, which indicates that no devices can connect to the instance.

### Context

- You must configure a whitelist upon the first use of an instance. After the whitelist is configured, the connection address of the instance is displayed on the **Basic Information** and **Database Connection** pages.
- Proper configuration of the whitelists can enhance access security of ApsaraDB for MongoDB. We recommend that you regularly maintain the whitelist.

### Procedure

1. Log on to the [ApsaraDB for MongoDB console](#).
2. In the upper-left corner of the page, select the resource group and the region of the target instance.
3. In the left-side navigation pane, click **Sharded Cluster Instances**.
4. Find the target instance and click its ID.
5. In the left-side navigation pane, choose **Data Security > Whitelist Settings**.
6. Click the  icon in the **Actions** column, and select **Manually Modify** or **Import ECS Intranet IP**.
  - Click **Manually Modify**. In the dialog box that appears, enter an IP address or CIDR block, and click **OK**.
  - Click **Import ECS Intranet IP**. In the dialog box that appears, the internal IP addresses of the ECS instances of your Alibaba Cloud account are displayed. You can select the desired IP addresses, add them to a whitelist, and click **OK**.

#### Note

- If a whitelist contains more than one IP address, separate them with commas (,). Every IP address in a whitelist must be unique. A whitelist can contain a maximum of 1,000 IP addresses.  
  
Supported formats include 0.0.0.0/0, 10.23.12.24 (single IP address), and 10.23.12.24/24. 10.23.12.24/24 is a CIDR notation (for more information, see [CIDR blocks](#)), in which the suffix /24 indicates the number of bits for the prefix of the IP address. The prefix consists of 1 to 32 bits.
- If the value is 0.0.0.0/0 or empty, the ApsaraDB for MongoDB instance can be accessed by all IP addresses. In this situation, the database is at high security risk.

### Related operations

- [Add an ECS security group for a standalone, replica set, or sharded cluster instance](#)

- [Delete an IP whitelist or an ECS security group of a standalone, replica set, or sharded cluster instance](#)

## Common connection scenarios

- [Connect a local client to an ApsaraDB for MongoDB instance over the Internet](#)
- [How to connect an ECS instance to an ApsaraDB for MongoDB instance when their network types are different](#)
- [How to connect an ECS instance to an ApsaraDB for MongoDB instance when they are in different regions](#)
- [How to connect an ECS instance to an ApsaraDB for MongoDB instance when they do not belong to the same Alibaba Cloud account](#)

# 7. Apply for a public endpoint for a sharded cluster instance

This topic describes how to apply for a public endpoint for a sharded cluster instance when you want to connect to this instance over the Internet.

## Context

The following table describes the connections supported by ApsaraDB for MongoDB.

| Connection type                       | Description  |
|---------------------------------------|--|
| Intranet Connection - VPC             | <ul style="list-style-type: none"> <li>A VPC is an isolated network with higher security and performance than a classic network.</li> <li>By default, ApsaraDB for MongoDB provides endpoints on a VPC.</li> </ul>   |
| Intranet Connection - Classic Network | Cloud services on a classic network are not isolated. Unauthorized access can only be blocked by using security groups or whitelists. You can switch the network type to VPC. For more information, see <a href="#">Switch the network type of an ApsaraDB for MongoDB instance</a> .  |
| Public IP Connection                  | <ul style="list-style-type: none"> <li>Connecting to a replica set instance over the Internet is risky. Therefore, ApsaraDB for MongoDB does not provide public endpoints.</li> <li>If you want to connect to a replica set instance from a device outside Alibaba Cloud (for example, a local client), you must apply for a public endpoint.</li> </ul> |

## Procedure

- Log on to the [ApsaraDB for MongoDB console](#).
- In the upper-left corner of the page, select the resource group and the region of the target instance.
- In the left-side navigation pane, click **Sharded Cluster Instances**.
- Find the target instance and click its ID.
- In the left-side navigation pane, click **Database Connection**.
- In the upper-right corner of the **Public IP Connection** section, click **Apply for Public Connection String**.

- In the dialog box that appears, specify **Node Type** and **Node ID**, and click **OK**.

| Parameter | Value | Description |
|-----------|-------|-------------|
|           |       |             |

| Parameter | Value  | Description   |
|-----------|--|---|
| Node Type | shard  | <p>A shard. Before you apply for a public endpoint for a shard, you must apply for an internal endpoint for it. For more information, see <a href="#">Apply for a connection string of a shard or Configserver node</a>.</p> <p>If you want to read the oplog data of a shard over the Internet when you perform certain operations such as data synchronization between clusters, you must apply for a public endpoint for the shard.</p>  |
|           | cs   | <p>The config server. Before you apply for a public endpoint for the config server, you must apply for an internal endpoint for it. For more information, see <a href="#">Apply for a connection string of a shard or Configserver node</a>.</p> <p>If you want to read the configuration information of the config server over the Internet when you perform certain operations such as data synchronization between clusters, you must apply for a public endpoint for the config server.</p> |
|           | mongos   | A mongos. This is the default option because your application is connected to a mongos in most cases.   |
| Node ID   | The ID of the component for which you want to apply for a public endpoint. | None  |

#### Note

- For more information about component types, see [Architecture of sharded cluster instances](#).
- To apply for a public endpoint for other mongos, repeat this step. You can only apply for a new public endpoint after the current one is created.

## Results

When the application is complete, the sharded cluster instance generates new endpoints for both the primary and secondary nodes and the corresponding connection string URI. For more information, see [Overview of sharded cluster instance connections](#).

## References

- [Connect a local client to an ApsaraDB for MongoDB instance over the Internet](#).
- To ensure data security, we recommend that you release a public endpoint if you no longer need it. For more information, see [Release a public connection string](#).

## What's next

Before you connect to the sharded cluster instance by using one of the obtained public endpoints, you must add the public IP address of your client to a whitelist of the instance. For more information, see

Configure a whitelist for a sharded cluster instance.

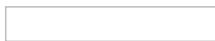
# 8. Connect to an instance

## 8.1. Overview of sharded cluster instance connections

ApsaraDB for MongoDB supports both connection strings and connection string URIs. You can use a connection string to connect to one mongos, and use a connection string URI to connect to more mongos. For high availability, we recommend that you use connection string URIs to connect your application to more mongos. This topic provides an overview of sharded cluster instance connections.

### View connection strings

1. Log on to the [ApsaraDB for MongoDB console](#).
2. In the upper-left corner of the page, select the resource group and the region of the target instance.
3. In the left-side navigation pane, click **Sharded Cluster Instances**.
4. Find the target instance and click its ID.
5. In the left-side navigation pane, click **Database Connection** to view connection strings.



### Description of connection strings

| Item      | Description   |
|-----------|---|
| Type      | <ul style="list-style-type: none"> <li>• Intranet Connection - Classic Network: Cloud services on a classic network are not isolated. Unauthorized access can only be blocked by using security groups or whitelists.</li> <li>• Intranet Connection - VPC: A VPC is an isolated network with higher security and performance than a classic network. By default, an ApsaraDB for MongoDB instance provides VPC connection addresses.</li> <li>• Public IP Connection: Connecting to a sharded cluster instance over the Internet is risky. Therefore, ApsaraDB for MongoDB does not provide public endpoints. If you want to connect to a sharded cluster instance from a device outside Alibaba Cloud (for example, a local client), you must apply for a public endpoint. For more information, see <a href="#">Apply for a public endpoint for a sharded cluster instance</a>.</li> </ul> |
| mongos ID | <p>The connection string of a mongos is in the following format:</p> <pre>&lt;host&gt;:&lt;port&gt;</pre> <ul style="list-style-type: none"> <li>• &lt;host&gt;: the endpoint used to connect to the sharded cluster instance.</li> <li>• &lt;port&gt;: the port used to connect to the sharded cluster instance.</li> </ul> <p><b>Note</b> During regular tests, you can use a connection string to directly connect to a mongos.</p>  |

| Item                  | Description   |
|-----------------------|---|
| Connection string URI | <p>A connection string URI is in the following format:</p> <pre>mongodb://[username:password@]host1[:port1][,host2[:port2],...[,hostN[:portN]]]/[database][?options]]</pre> <ul style="list-style-type: none"> <li>• <code>mongodb://</code>: the prefix, indicating a connection string URI.</li> <li>• <code>username:password@</code>: the username and password used to log on to a database of the sharded cluster instance. You must separate the username and password with a colon (:).</li> <li>• <code>hostX:portX</code>: the endpoint and port of a mongos in the sharded cluster instance.</li> <li>• <code>/database</code>: the database corresponding to the username and password if authentication is enabled.</li> <li>• <code>?options</code>: additional connection options.</li> </ul> <p><b>Note</b> If your application is in a production environment, we recommend that you use a connection string URI to connect to the instance. Then your client can automatically distribute your requests to multiple mongos to balance loads. When a mongos fails, your client automatically redirects requests to other mongos in the normal state.</p> |

## Log on to a database of the sharded cluster instance

1. Obtain the [connection strings](#) and the following information:

- The username used to log on to the database. The initial username is root.

**Note** We recommend that you do not log on to a database as the root user in a production environment. You can create accounts and grant permissions to them based on your needs. For more information, see [Manage MongoDB users through DMS](#).

- The password of the database user. If you forget the password of the root user, you can reset it. For more information, see [Set a password for a sharded cluster instance](#).
- The name of the database corresponding to the username if authentication is enabled. If the username is root, enter admin.

2. Log on to the database.

- [Connect to a sharded cluster ApsaraDB for MongoDB instance through DMS](#)
- [Connect to a sharded cluster instance by using the mongo shell](#)
- [Connect to an ApsaraDB for MongoDB instance through the program code](#)

## Common connection scenarios

- [Connect a local client to an ApsaraDB for MongoDB instance over the Internet](#)
- [How to connect an ECS instance to an ApsaraDB for MongoDB instance when their network types are different](#)
- [How to connect an ECS instance to an ApsaraDB for MongoDB instance when they are in different regions](#)
- [How to connect an ECS instance to an ApsaraDB for MongoDB instance when they do not belong to the same Alibaba Cloud account](#)

## FAQ


- [How to troubleshoot logon issues for the mongo shell](#)
- [How to troubleshoot database connection failures after the number of connections reaches the upper limit](#)
- [How to troubleshoot the high CPU utilization of ApsaraDB for MongoDB](#)
- [How to query and limit the number of connections](#)

## 8.2. Connect to a sharded cluster ApsaraDB for MongoDB instance through DMS

Data Management (DMS) is an integrated database solution that offers data management, structure management, user authorization, security auditing, data trend analysis, data tracking, BI charts, performance optimization, and server management. You can use DMS to connect to a sharded cluster ApsaraDB for MongoDB instance for easy management.

### Preparations

Add the IP address of the DMS server to the whitelist of the ApsaraDB for MongoDB instance based on the network type. For more information, see [Configure a whitelist for a sharded cluster instance](#).

 **Note** Skip this step if you have added the IP address of the DMS server to the whitelist of the ApsaraDB for MongoDB instance.

### IP addresses of the DMS server

| Network type of ApsaraDB for MongoDB instance | IP address of the DMS server |
|---|------------------------------|
| VPC   | 100.104.0.0/16               |
| Classic network                               | 120.55.177.0/24              |
|   | 121.43.18.0/24               |
|   | 101.37.74.0/24               |
|   | 10.153.176.0/24              |
|   | 10.137.42.0/24               |
|   | 11.193.54.0/24               |

### Procedure

1. Log on to the [ApsaraDB for MongoDB console](#).
2. In the upper-left corner of the page, select the resource group and the region of the target instance.
3. In the left-side navigation pane, click **Sharded Cluster Instances**.
4. Find the target instance and click its ID.
5. Click **Log On** and select any Mongos node ID in the upper-right corner of the Basic Information page. You are redirected to the **DMS** console.





6. In the DMS console, enter the following information.

| Item                        | Description  |
|-----------------------------|--|
| <b>Network address:Port</b> | The internal connection string of the Mongos node of the ApsaraDB for MongoDB instance is automatically entered.   |
| <b>Database Username</b>    | Enter the database account of the MongoDB instance. The initial account is root.   |
| <b>Database Name</b>        | Enter the name of the database to which the account belongs.<br><br><div style="background-color: #e6f2ff; padding: 10px; border: 1px solid #d9e1f2;"> <p><b>Note</b></p> <ul style="list-style-type: none"> <li>◦ If <b>Database Username</b> is set to root, the database name is admin.</li> <li>◦ We do not recommend that you log on to a database as the root user in the production environment. You can create users and grant permissions based on your business needs. For more information, see <a href="#">Use DMS to manage ApsaraDB for MongoDB users</a>.</li> </ul> </div> |
| <b>Password</b>             | The password of the specified account.<br><br><div style="background-color: #e6f2ff; padding: 10px; border: 1px solid #d9e1f2;"> <p><b>Note</b> If you forget the password of the root account, you can reset the password by using the method specified in <a href="#">Set a password</a>.</p> </div>   |

7. Click **Log On**.

## Common connection scenarios

- [Connect a local client to an ApsaraDB for MongoDB instance over the Internet](#)
- [How to connect an ECS instance to an ApsaraDB for MongoDB instance when their network types are different](#)
- [How to connect an ECS instance to an ApsaraDB for MongoDB instance when they are in different regions](#)
- [How to connect an ECS instance to an ApsaraDB for MongoDB instance when they do not belong to the same Alibaba Cloud account](#)

## FAQ


- [How to troubleshoot logon issues for the mongo shell](#)
- [How to troubleshoot database connection failures after the number of connections reaches the upper limit](#)
- [Troubleshoot high CPU utilization of ApsaraDB for MongoDB](#)
- [How to query and limit the number of connections](#)

## 8.3. Connect to a sharded cluster instance by using the mongo shell

This topic describes how to connect to a sharded cluster instance by using the mongo shell, which is a database management tool provided with MongoDB. You can install the mongo shell on your client or an ECS instance.

## Prerequisites

- To ensure successful authentication, the version of the mongo shell must match with that of the ApsaraDB for MongoDB instance. For more information about the installation procedure, visit [Install MongoDB](#). Select the correct version based on your client.
- The IP address of your client is added to a whitelist of the sharded cluster instance. For more information, see [Configure a whitelist for a sharded cluster instance](#).

 **Note** If you want to connect to the instance over the Internet, you must [apply for a public endpoint](#).

## Procedure

1. Log on to the [ApsaraDB for MongoDB console](#).
2. In the upper-left corner of the page, select the resource group and the region of the target instance.
3. In the left-side navigation pane, click **Sharded Cluster Instances**.
4. Find the target instance and click its ID.
5. In the left-side navigation pane, click **Database Connection** to obtain the connection string of a mongos.

6. Connect to the sharded cluster instance from your client or ECS instance that has the mongo shell installed.

```
mongo --host <mongos_host> -u <username> -p --authenticationDatabase <database>
```


### Note

- <mongos\_host>: the connection string of a mongos in the sharded cluster instance.
- <username>: the username used to log on to a database of the sharded cluster instance. The initial username is root. We recommend that you do not log on to a database as the root user in a production environment. You can create accounts and grant permissions to them as needed. For more information, see [Manage MongoDB users through DMS](#).
- <database>: the name of database corresponding to the username if authentication is enabled. If the username is root, enter admin.

Example:

```
mongo --host s-bp*****.mongodb.rds.aliyuncs.com:3717 -u root -p --authenticationDatabase admin
```

7. When **Enter password:** is displayed, enter the password of the database user and press Enter. If you forget the password of the root user, you can reset it. For more information, see [Set a password for a sharded cluster instance](#).

 **Note** The password characters are not displayed when you enter the password.

## Common connection scenarios

- [Connect a local client to an ApsaraDB for MongoDB instance over the Internet](#)
- [How to connect an ECS instance to an ApsaraDB for MongoDB instance when their network types are different](#)
- [How to connect an ECS instance to an ApsaraDB for MongoDB instance when they are in different regions](#)
- [How to connect an ECS instance to an ApsaraDB for MongoDB instance when they do not belong to the same Alibaba Cloud account](#)


## FAQ

- [How to troubleshoot logon issues for the mongo shell](#)
- [How to troubleshoot database connection failures after the number of connections reaches the upper limit](#)
- [How to troubleshoot the high CPU utilization of ApsaraDB for MongoDB](#)
- [How to query and limit the number of connections](#)

# 8.4. Connect to an ApsaraDB for MongoDB instance through the program code

## Related links

- [MongoDB Drivers](#)
- [Connection String URI Format](#)

 **Note** The connection sample code in this topic applies when you use internal IP addresses provided by Alibaba Cloud to connect to ApsaraDB for MongoDB.

- For more information about how to obtain connection strings of ApsaraDB for MongoDB, see [Connect to an ApsaraDB for MongoDB instance](#).

## Node.js

Related links: [MongoDB Node.js Driver](#)

1. Initialize a project.

```
mkdir node-mongodb-demo
cd node-mongodb-demo
npm init
```

2. Install the driver package and toolkit.

```
npm install mongodb node-uuid sprintf-js -save
```

3. Obtain connection strings of ApsaraDB for MongoDB instances.
4. Use the following Node.js sample code.

```
'use strict';
```

```
var uuid = require('node-uuid');
var sprintf = require("sprintf-js").sprintf;
var MongoClient = require('mongodb').MongoClient;
var host1 = "demotest-1.mongodb.tbc3.newtest.rdstest.aliyun-inc.com";
var port1 = 27017;
var host2 = "demotest-2.mongodb.tbc3.newtest.rdstest.aliyun-inc.com";
var port2 = 27017;
var username = "demouser";
var password = "123456";
var replSetName = "mgset-1441984991";
var demoDb = "test";
var demoColl = "testColl";
// The officially recommended solution.
var url = sprintf("mongodb://%s:%d,%s:%d/%s? replicaSet=%s", host1, port1, host2, port2, demoDb, replSetName);
console.info("url:", url);
// Obtain the MongoClient.
mongoClient.connect(url, function(err, db) {
  if(err) {
    console.error("connect err:", err);
    return 1;
  }
  // Authenticate. Here, the username is for authentication of the admin database.
  var adminDb = db.admin();
  adminDb.authenticate(username, password, function(err, result) {
    if(err) {
      console.error("authenticate err:", err);
      return 1;
    }
    // Obtain the collection handle.
    var collection = db.collection(demoColl);
    var demoName = "NODE:" + uuid.v1();
    var doc = {"DEMO": demoName, "MESG": "Hello AliCoudDB For MongoDB"};
    console.info("ready insert document: ", doc);
    // Insert data.
    collection.insertOne(doc, function(err, data) {
      if(err) {
        console.error("insert err:", err);
        return 1;
      }
      console.info("insert result:", data["result"]);
      // Read data
```

```
// Read data
var filter = {"DEMO": demoName};
collection.find(filter).toArray(function(err, items) {
  if(err) {
    console.error("find err:", err);
    return 1;
  }
  console.info("find document: ", items);
  // Close the client and release resources.
  db.close();
});
});
});
});
```

## PHP

Related links:

[Mongodb php driver](#)

1. Install the driver package and toolkit.

```
$ pecl install mongodb
$ echo "extension=mongodb.so" >> `php --ini | grep "Loaded Configuration" | sed -e "s|.*:|s*||" `
$ composer require "mongodb/mongodb=^1.0.0"
```

2. Obtain connection strings of ApsaraDB for MongoDB instances.
3. Use the following PHP sample code.

```
<? php
require 'vendor/autoload.php'; // include Composer goodies

# Instance information
$demo_seed1 = 'demotest-1.mongodb.test.aliyun-inc.com:3717';
$demo_seed2 = 'demotest-2.mongodb.test.aliyun-inc.com:3717';
$demo_replname = "mgset-1441984463";
$demo_user = 'root';
$demo_password = '123456';
$demo_db = 'admin';

# Construct the mongodb connection string based on the instance information.
# mongodb://[username:password@]host1[:port1],[host2[:port2],...[,hostN[:portN]]][/[database][? options]]

$demo_uri = 'mongodb://' . $demo_user . ':' . $demo_password . '@' .
    $demo_seed1 . ',' . $demo_seed2 . '/' . $demo_db . '? replicaSet=' . $demo_replname;
$client = new MongoClient($demo_uri);
$collection = $client->testDb->testColl;
$result = $collection->insertOne( [ 'name' => 'ApsaraDB for MongoDB', 'desc' => 'Hello, MongoDB' ] );
echo "Inserted with Object ID '{$result->getInsertedId()}'", "\n";
$result = $collection->find( [ 'name' => 'ApsaraDB for MongoDB' ] );
foreach ($result as $entry)
{
    echo $entry->_id, ':', $entry->name, "\n";
}
?>
```

## Java

Related links:

- Official [Quick Start](#)
- JAR package [download](#)

1. Obtain connection strings of ApsaraDB for MongoDB instances.
2. Use the following Java sample code.
  - o Maven configuration

```
<dependencies>
  <dependency>
    <groupId>org.mongodb</groupId>
    <artifactId>mongo-java-driver</artifactId>
    <version>3.0.4</version>
  </dependency>
</dependencies>
```

## ◦ Java sample code

```
import java.util.ArrayList;
import java.util.List;
import java.util.UUID;
import org.bson.BsonDocument;
import org.bson.BsonString;
import org.bson.Document;
import com.mongodb.MongoClient;
import com.mongodb.MongoClientOptions;
import com.mongodb.MongoClientURI;
import com.mongodb.MongoCredential;
import com.mongodb.ServerAddress;
import com.mongodb.client.MongoCollection;
import com.mongodb.client.MongoCursor;
import com.mongodb.client.MongoDatabase;
public class Main {
    public static ServerAddress seed1 = new ServerAddress("demotest-1.mongodb.tbc3.newtest.rdstes
t.aliyun-inc.com", 27017);
    public static ServerAddress seed2 = new ServerAddress("demotest-2.mongodb.tbc3.newtest.rdstes
t.aliyun-inc.com", 27017);
    public static String username = "demouser";
    public static String password = "123456";
    public static String ReplSetName = "mgset-1441984463";
    public static String DEFAULT_DB = "admin";
    public static String DEMO_DB = "test";
    public static String DEMO_COLL = "testColl";
    public static MongoClient createMongoDBClient() {
        // Construct a seed list.
        List<ServerAddress> seedList = new ArrayList<ServerAddress>();
        seedList.add(seed1);
        seedList.add(seed2);
        // Construct authentication information.
        List<MongoCredential> credentials = new ArrayList<MongoCredential>();
        credentials.add(MongoCredential.createScramSha1Credential(username,
            DEFAULT_DB, password.toCharArray()));
        // Construct operation options. Configure options other than requiredReplicaSetName based on y
our actual requirements. The default parameter settings are sufficient for most scenarios.
        MongoClientOptions options = MongoClientOptions.builder()
            .requiredReplicaSetName(ReplSetName).socketTimeout(2000)
            .connectionsPerHost(1).build();
        return new MongoClient(seedList, credentials, options);
    }
}
```

```
}
public static MongoClient createMongoDBClientWithURI() {
    // Use a URI to initialize the MongoClient.
    //mongodb://[username:password@]host1[:port1][,host2[:port2],...[,hostN[:portN]]][/[database][
? options]]
    MongoClientURI connectionString = new MongoClientURI("mongodb://" + username + ":" + password + "@" +
        seed1 + "," + seed2 + "/" +
        DEFAULT_DB +
        "? replicaSet=" + ReplSetName);
    return new MongoClient(connectionString);
}
public static void main(String args[]) {
    MongoClient client = createMongoDBClient();
    //or
    //MongoClient client = createMongoDBClientWithURI();
    try {
        // Obtain the collection handle.
        MongoDB database = client.getDatabase(DEMO_DB);
        MongoCollection<Document> collection = database.getCollection(DEMO_COLL);
        // Insert data.
        Document doc = new Document();
        String demoname = "JAVA:" + UUID.randomUUID();
        doc.append("DEMO", demoname);
        doc.append("MSG", "Hello AliCoudDB For MongoDB");
        collection.insertOne(doc);
        System.out.println("insert document: " + doc);
        // Read data.
        BsonDocument filter = new BsonDocument();
        filter.append("DEMO", new BsonString(demoname));
        MongoCursor<Document> cursor = collection.find(filter).iterator();
        while (cursor.hasNext()) {
            System.out.println("find document: " + cursor.next());
        }
    } finally {
        // Close the client and release resources.
        client.close();
    }
    return ;
}
}
```



## Python

Related links:

- [Pymongo download](#)
- [Official documentation](#)

1. Install PyMongo.

```
pip install pymongo
```

2. Obtain the connection strings of ApsaraDB for MongoDB instances.
3. Use the following Python sample code.

```
import uuid
from pymongo import MongoClient
# Specify two addresses used to connect to the primary and secondary nodes of the instance.
CONN_ADDR1 = 'demotest-1.mongodb.tbc3.newtest.rdstest.aliyun-inc.com:27017'
CONN_ADDR2 = 'demotest-2.mongodb.tbc3.newtest.rdstest.aliyun-inc.com:27017'
REPLICAT_SET = 'mgset-1441984463'
username = 'demouser'
password = '123456'
# Obtain the MongoClient.
client = MongoClient([CONN_ADDR1, CONN_ADDR2], replicaSet=REPLICAT_SET)
# Authenticate. Here, the username is for authentication of the admin database.
client.admin.authenticate(username, password)
# Use the collection:testColl of the test database as an example. Insert doc and search for documents based on the demo name.
demo_name = 'python-' + str(uuid.uuid1())
print 'demo_name:', demo_name
doc = dict(DEMO=demo_name, MESSG="Hello ApsaraDB For MongoDB")
doc_id = client.test.testColl.insert(doc)
print 'doc_id:', doc_id
for d in client.test.testColl.find(dict(DEMO=demo_name)):
    print 'find documents:', d
```

# 9. Migrate data


## 9.1. Migrate a user-created sharded MongoDB database to ApsaraDB for MongoDB by using DTS

This topic describes how to migrate shards of a user-created sharded MongoDB database to ApsaraDB for MongoDB by using Data Transmission Service (DTS). DTS allows you to migrate historical and incremental data without service disruptions.

For more information about data migration and synchronization solutions, see [Overview of data migration and synchronization](#).


### Prerequisite

- The version of the user-created MongoDB database is 3.0, 3.2, 3.4, 3.6, or 4.0.
- Each shard in the destination sharded cluster instance has sufficient storage space.

 **Note** For example, a user-created MongoDB database has three shards, and one of these shards occupies the maximum storage space of 500 GB. In this case, the storage space of each shard in destination instance must be greater than 500 GB.

### How it works

DTS migrates a user-created MongoDB database by migrating each shard in the instance. You must create a data migration task for each shard.

 **Note** The distribution of migrated data in the destination instance depends on the shard key that you specify. For more information, see [Configure sharding to maximize the performance of shards](#).



### Usage notes

- DTS uses resources of the source and destination instances during full data migration. This may increase the load of the database server. If the data volume is large or the specification is low, the database server may become unavailable. We recommend that you migrate user-created MongoDB databases during off-peak hours.
- If the source user-created MongoDB database and the destination ApsaraDB for MongoDB instance run different MongoDB versions or storage engines, ensure that your applications can run on both instances. For more information about MongoDB versions and storage engines that are supported by ApsaraDB for MongoDB, see [MongoDB versions and storage engines](#).


### Billing information

| Migration type | Instance fee | Internet traffic fee |
|----------------|--------------|----------------------|
|----------------|--------------|----------------------|


| Migration type             | Instance fee   | Internet traffic fee   |
|----------------------------|--|--|
| Full data migration        | Free of charge   | Charged only when data is migrated from Alibaba Cloud over the Internet. For more information, see <a href="#">Data Transmission Service Pricing</a> . |
| Incremental data migration | Charged. For more information, see <a href="#">Data Transmission Service Pricing</a> . |  |

## Migration types

- Full data migration: All historical data in the source instance is migrated to the destination instance.

 **Note** Data migration is supported at the database, collection, and index levels.

- Incremental data migration: After full data migration, incremental data is synchronized to the destination instance.

 **Note**

- The create and delete operations on databases, collections, and indexes can also be synchronized.
- The create, delete, and update operations on documents can be synchronized.

## Required database account permissions

| Data source                               | Support for full data migration                    | Support for incremental data migration                                      |
|---|--|---|
| Source user-created MongoDB database      | Read permissions on the source database            | Read permissions on the source database, admin database, and local database |
| Destination ApsaraDB for MongoDB instance | Read/write permissions on the destination database | Read/write permissions on the destination database                          |

How to create and authorize a database account:

- For a user-created MongoDB database, see [db.createUser\(\)](#).
- For an ApsaraDB for MongoDB instance, see [Manage MongoDB users through DMS](#).

## Prerequisites

1. To avoid the impact of block migration on data consistency, you must disable the balancer of the user-created MongoDB databases during migration. For more information, see [Manage the ApsaraDB for MongoDB balancer](#).

 **Warning** If the balancer is not disabled, block migration will affect the consistency of the data read by DTS.

2. Delete the orphaned documents generated due to migration failures from the user-created MongoDB database.

**Note** If the orphaned documents are not deleted, the documents with `_id` conflicts may exist during migration and unwanted data may be migrated.

- i. Download the `cleanupOrphaned.js` file.

```
wget "http://docs.aliyun.cn-hangzhou.oss.aliyun-inc.com/assets/attach/120562/cn_zh/1564451237979/cleanupOrphaned.js"
```

- ii. Replace `test` in the `cleanupOrphaned.js` file with the name of the database where you want to delete orphaned documents.

**Note** If you want to delete orphaned documents from multiple databases, repeat Steps ii and iii.

- iii. Run the following command on a shard to delete the orphaned documents from all collections in the specified database:

**Note** You must repeat this step on each shard.

```
mongo --host <Shardhost> --port <Primaryport> --authenticationDatabase <database> -u <username> -p <password> cleanupOrphaned.js
```

**Note**

- `<Shardhost>`: the IP address of the shard.
- `<Primaryport>`: the service port of the primary node of the shard.
- `<database>`: the database corresponding to the username and password if authentication is enabled.
- `<username>`: the account used to log on to the user-created MongoDB database.
- `<password>`: the password used to log on to the user-created MongoDB database.

The following figure shows an example.

In this example, a user-created MongoDB database has three shards, and you must delete the orphaned documents on each of the shards.


```
mongo --host 172.16.1.10 --port 27018 --authenticationDatabase admin -u root -p 'Test123456' cleanupOrphaned.js
```

```
mongo --host 172.16.1.11 --port 27021 --authenticationDatabase admin -u root -p 'Test123456' cleanupOrphaned.js
```

```
mongo --host 172.16.1.12 --port 27024 --authenticationDatabase admin -u root -p 'Test123456' cleanupOrphaned.js
```



3. Create required databases and collections in the destination sharded cluster instance, and configure

data sharding for the databases and collections. For more information, see [Configure sharding to maximize the performance of shards](#).

 **Note** If you configure data sharding before you start data migration, data in the user-created MongoDB database is evenly migrated to the shards in the destination sharded cluster instance. This prevents overloading a single shard.

## Procedure

1. Log on to the [Data Transmission Service console](#).
  2. In the left-side navigation pane, click **Data Migration**.
  3. In the **Migration Tasks** section, select the region in which the ApsaraDB for MongoDB instance is deployed.
- 
4. In the upper-right corner, click **Create Migration Task**.
  5. Click **Create Migration Task**. In the **Configure Source and Destination** step, configure the source and destination databases for the migration task.

| Section | Parameter       | Description   |
|---------|-----------------|---|
| N/A     | Task Name       | DTS automatically generates a task name. We recommend that you specify an informative name for easy identification. You do not need to use a unique task name.  |
|         | Instance Type   | Select an instance type based on the location where the database is deployed. In this topic, a <b>User-Created Database with Public IP Address</b> is used as an example. <div style="background-color: #e0f2f1; padding: 5px; margin-top: 10px;">  <b>Note</b> If you select other instance types, you must prepare the environment that is required for the source database. For more information, see <a href="#">Preparation overview</a>.                     </div>  |
|         | Instance Region | If Instance Type is set to <b>User-Created Database with Public IP Address</b> , you do not need to specify the <b>Instance Region</b> . <div style="background-color: #e0f2f1; padding: 5px; margin-top: 10px;">  <b>Note</b> If you have configured a whitelist for the user-created MongoDB database, you must add the CIDR blocks of DTS servers to the whitelist. You can click <a href="#">Get IP Address Segment of DTS</a> next to <b>Instance Region</b> to obtain the CIDR blocks of DTS servers.                     </div> |
|         | Database Type   | Select <b>MongoDB</b> .   |

| Section         | Parameter              | Description  |
|-----------------|------------------------|--|
| Source Database | Hostname or IP Address | <p>Enter the endpoint of a shard for the source database. In this example, enter the public IP address of the shard.</p> <p><b>Note</b> DTS migrates each shard of the source database in turn. In this example, enter the endpoint of the first shard. Then enter the endpoint of the second shard in the second migration task. Repeat this until all shards are migrated.</p>   |
|                 | Port Number            | <p>Enter the service port of the shard.</p> <p><b>Note</b> The service port of each shard for user-created MongoDB database must be open to the public network.</p>  |
|                 | Database Name          | Enter the name of the authentication database to which the database account belongs.   |
|                 | Database Account       | Enter the username of the database account used to manage the source database. For more information about the permissions that are required for the account, see <a href="#">Required database account permissions</a> .   |
|                 | Database Password      | <p>Enter the password of the destination database account.</p> <p><b>Note</b> After you specify the source database information, click <b>Test Connectivity</b> next to <b>Database Password</b> to check whether the information is correct. If the information is correct, the <b>Passed</b> message is displayed. If the <b>Failed</b> message is displayed, click <b>Check</b> in the <b>Failed</b> message to modify the information as prompted.</p> |
|                 | Encryption             | <p>Select <b>Non-encrypted</b>.</p> <p><b>Note</b> The <b>SSL-encrypted</b> option is available only when you migrate MongoDB Atlas.</p>   |
|                 | Instance Type          | The type of the instance. In this example, select <b>MongoDB Instance</b> .  |
|                 | Instance Region        | The region where the ApsaraDB for MongoDB instance resides.  |
|                 | MongoDB Instance ID    | Select the ID of the ApsaraDB for MongoDB instance.  |

| Section              | Parameter         | Description  |
|----------------------|-------------------|--|
| Destination Database | Database Name     | Enter the name of the authentication database to which the database account belongs.<br><br><b>Note</b> If you want to use the root account, specify admin for the Database Name parameter.  |
|                      | Database Account  | Enter the username of the database account used to manage the source database. For more information about the permissions that are required for the account, see <a href="#">Required database account permissions</a> .   |
|                      | Database Password | Enter the password of the destination database account.<br><br><b>Note</b> After you specify the destination database information, click <b>Test Connectivity</b> next to <b>Database Password</b> to check whether the information is correct. If the information is correct, the <b>Passed</b> message is displayed. If the information is incorrect, the <b>Failed</b> message is displayed, and you must click <b>Check</b> next to the <b>Failed</b> message to modify the information as prompted. |

6. In the lower-right corner of the page, click **Set Whitelist and Next**.

**Note** The CIDR blocks of DTS servers are automatically added to the whitelist of the destination ApsaraDB for MongoDB instance. This ensures that DTS servers can connect to the destination ApsaraDB for MongoDB instance. After the migration is completed, you can remove these CIDR blocks from the whitelist. For more information, see [Configure a whitelist or an ECS security group for an ApsaraDB for MongoDB instance](#).

7. Select the migration types and objects to be migrated.



8. In the lower-right corner of the page, click **Precheck**.

**Note**

- A precheck is performed before the migration task starts. You can start the data migration task only after the task passes the precheck.
- If the task fails to pass the precheck, click the  icon next to each failed item to view details. Troubleshoot the issues based on the causes and run the precheck again.

9. After the task passes the precheck, click **Next**.

10. In the **Confirm Settings** dialog box, specify the **Channel Specification** and select **Data**

**Transmission Service (Pay-As-You-Go) Service Terms.**


11. Click **Buy and Start** to start the migration task.
12. Repeat Steps 1 to 11 to create data migration tasks for the remaining shards.
13. Stop the data migration task.

- Full data migration

Do not manually stop a task during full data migration. Otherwise, the system may fail to perform a full data migration. Wait until the data migration task automatically stops.

- Incremental data migration

An incremental data migration task does not automatically stop. You must manually stop the migration task.

 **Note** Select an appropriate time to manually stop the migration task. For example, you can stop the migration task during off-peak hours or before you switch your workloads to the destination instance.

- a. Wait until **Incremental Data Migration** and **The migration task is not delayed** appear in the progress bar of the migration task. Then, stop writing data to the source database for a few minutes. The delay time of **incremental data migration** may be displayed in the progress bar.
- b. After the status of **Incremental Data Migration** changes to **The migration task is not delayed**, stop the migration task.

14. Switch over your business to the destination ApsaraDB for MongoDB instance.

## 9.2. Migrate a user-created MongoDB database to ApsaraDB for MongoDB by using tools provided by MongoDB

This topic describes how to migrate a user-created MongoDB database to ApsaraDB for MongoDB by using `mongodump` and `mongorestore`, which are built in MongoDB for data backup and restoration. You can install user-created MongoDB databases on a local server or an ECS instance, and use `mongodump` and `mongorestore` to migrate these databases to an ApsaraDB for MongoDB sharded cluster instance.


### Background information

- To avoid service disruption, we recommend that you use DTS to migrate user-created sharded MongoDB databases to ApsaraDB for MongoDB. For more information, see [Migrate a user-created sharded MongoDB database to ApsaraDB for MongoDB by using DTS](#).
- For more information about data migration and synchronization solutions, see [Overview of data migration and synchronization](#).

### Prerequisites

- `mongodump` and `mongorestore` are installed on a different server from the user-created MongoDB databases, but run the same version as the databases. For more information about the installation procedure, visit [Install MongoDB](#) at the MongoDB official website.



 **Note** You can also run the `mongodump` and `mongorestore` commands on the server where the user-created MongoDB databases reside.

- The storage capacity of the destination sharded cluster instance is greater than the storage space occupied by the user-created MongoDB databases. If the storage capacity is insufficient, you can upgrade the instance. For more information, see [Configuration change overview](#).

## Precautions

- This is full data migration. To ensure data consistency, we recommend that you stop writing data to the user-created MongoDB databases before you migrate data.
- If you have run the `mongodump` command to back up a user-created MongoDB database, move the backup files in the `dump` folder to another directory and make sure that the `dump` folder is empty. If it is not empty, its historical backup files are overwritten the next time you back up a database.
- Run the `mongodump` and `mongorestore` commands on the servers. Do not run these commands on the mongo shell.

## Step 1: Back up the user-created MongoDB databases

1. On the server where the user-created MongoDB databases reside, run the following command to back up all the databases:

```
mongodump --host <mongodb_host> --port <port> -u <username> --authenticationDatabase <database> >
```

### Note

- `<mongodb_host>`: the IP address of the server where the user-created MongoDB databases reside. In this case, enter `127.0.0.1`.
- `<port>`: the service port of the user-created MongoDB databases. The default value is `27017`.
- `<username>`: the username used to log on to a user-created MongoDB database.
- `<database>`: the database corresponding to the username if authentication is enabled.

Example:


```
mongodump --host 127.0.0.1 --port 27017 -u root --authenticationDatabase admin
```

2. When `Enter password:` is displayed, enter the password of the database user and press Enter. The data backup operation starts.

Wait until data backup is complete. The data of the user-created MongoDB databases is backed up to the `dump` folder of the directory where you run this command.


## Step 2: (Optional) Configure data sharding

If data sharding is not configured, data is only written to the primary shard. Then the storage and computing resources of other shards are not used. For more information, see [Configure sharding to maximize the performance of shards](#).


 **Note** You must create required databases and collections in the destination sharded cluster instance before data migration. However, you can configure data sharding for the databases and collections either before or after data migration.

### Step 3: Migrate data to the destination sharded cluster instance

1. Obtain the public or internal connection string of a mongos in the destination sharded cluster instance. For more information, see [Overview of sharded cluster instance connections](#).

 **Note** You must apply for a public endpoint manually. For more information, see [Apply for a public endpoint for a sharded cluster instance](#).

2. Add the IP address of the server where the user-created MongoDB databases reside to a whitelist of the destination sharded cluster instance. For more information, see [Configure a whitelist for a sharded cluster instance](#).

 **Note**

- If you want to connect to a sharded cluster instance over the internal network, you must add the private IP address of the ECS instance where the user-created MongoDB databases reside to a whitelist of the sharded cluster instance.
- If you want to connect to a sharded cluster instance over the Internet, you must add the public IP address of the server where the user-created MongoDB databases reside to a whitelist of the sharded cluster instance.

3. On the server where the user-created MongoDB databases reside, run the following command to restore all the backup files to the destination sharded cluster instance:

```
mongorestore --host <Mongos_host> -u <username> --authenticationDatabase <database> <Backup directory>
```

 **Note**

- <Mongos\_host>: the connection string of any mongos in the ApsaraDB for MongoDB instance.
- <username>: the username used to log on to a database of the destination sharded cluster instance. The initial username is root.
- <database>: the database corresponding to the username if authentication is enabled. If the username is root, enter admin.
- <Backup directory>: the directory where the backup files are stored. The default value is *dump*.

Example:

```
mongorestore --host s-bp*****-pub.mongodb.rds.aliyuncs.com:3717 -u root --authenticationDatabase admin dump
```

4. When **Enter password:** is displayed, enter the password of the database user and press Enter. The data restoration operation starts.

**Note**

- The password characters are not displayed when you enter the password.
- If you forget the password of the root user, you can reset it. For more information, see [Set a password for a sharded cluster instance](#).

After data restoration is complete, switch over your business to the destination sharded cluster instance. We recommend you perform the switchover during off-peak hours to minimize impact on your business.