Alibaba Cloud
ApsaraDB for MongoDB
Quick Start for Replica Set

Issue: 20200316
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 due to product version upgrades, adjustments, or other reasons. Alibaba Cloud reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through Alibaba Cloud-authorized channels. You shall 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 the document in the context that Alibaba Cloud products and services are provided on an "as is", "with all faults" and "as available" basis. 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 bear any liability for any errors or financial losses incurred by any organizations, companies, or individuals arising from their download, use, or trust in this document. Alibaba Cloud shall not, under any circumstances, bear responsibility for any indirect, consequent
ial, exemplary, incidental, 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 contact Alibaba Cloud directly if you discover any errors in this document.
# Document conventions

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

Legal disclaimer ....................................................................................................................... I
Document conventions .......................................................................................................... I
1 Before you start .................................................................................................................. 1
2 Get started with a replica set instance ............................................................................. 3
3 Create a replica set instance ............................................................................................. 4
4 Set a password for a replica set instance ......................................................................... 9
5 Configure a whitelist for a replica set instance ............................................................... 11
6 Apply for a public endpoint for a replica set instance .................................................... 15
7 Connect to an instance ..................................................................................................... 17
   7.1 Overview of replica set instance connections ............................................................. 17
   7.2 Connect to a replica set ApsaraDB for MongoDB instance through DMS .............. 20
   7.3 Connect to a replica set instance by using the mongo shell ................................ ...... 23
   7.4 Connection sample code for MongoDB drivers ......................................................... 27
8 Migrate data ....................................................................................................................... 35
   8.1 Migrate user-created MongoDB databases to Alibaba Cloud by using DTS .......... 35
   8.2 Migrate user-created MongoDB databases to Alibaba Cloud by using the built-in commands of MongoDB ............................................................... 42
1 Before you start

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

<table>
<thead>
<tr>
<th>Operation</th>
<th>Limit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Deploy a replica set instance</td>
<td>The database version must match the storage engine. For more information, see #unique_4.</td>
</tr>
</tbody>
</table>
| Build replica set nodes    | · A replica set instance automatically created by ApsaraDB for MongoDB consists of a primary node, a hidden secondary node (invisible to you), and one or more secondary nodes.  
                            · While a replica set instance is running, you can scale the instance to 3, 5, or 7 nodes as needed. For more information, see #unique_5. |
| Restart a replica set instance | You must restart the instance in the ApsaraDB for MongoDB console or by calling the #unique_7 operation. |
| Migrate data from a replica set instance | You can use the built-in commands of MongoDB or Data Transmission Service (DTS) to migrate data. For more information, see Migrate user-created MongoDB databases to Alibaba Cloud by using DTS or Migrate user-created MongoDB databases to Alibaba Cloud by using the built-in commands of MongoDB. |

Note:
You cannot connect to a replica set instance from a secondary node of a user-created database. If you want to synchronize data from a replica set instance to a user-created MongoDB database for data testing or analysis, you can use MongoShake. For more information, see #unique_6.
### Quick Start for Replica Set / 1 Before you start

<table>
<thead>
<tr>
<th>Operation</th>
<th>Limit</th>
</tr>
</thead>
</table>
| Back up the data of a replica set instance | - **Configure automatic backup for an ApsaraDB for MongoDB instance**: Only physical backup is supported.  
- **Manually back up an ApsaraDB for MongoDB instance**: Both physical backup and logical backup are supported. |
| Restore the data of a replica set instance | - You can restore data from a point in time or a backup set. For more information, see #unique_13 and #unique_14.  
- You can restore data to your current replica set instance only if it has three nodes. For more information, see #unique_15. |
| Modify parameters of a replica set instance | For security and stability, you are not allowed to modify certain parameters of a replica set instance. For more information, see #unique_16. |

**Note:**
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 #unique_12. Alternatively, you can select the database version 4.0 when you create the instance.
2 Get started with a replica set instance

This topic describes how to get started with a replica set 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 Before you start.

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

1. Create a replica set instance.
2. Set a password for a replica set instance.
3. Configure a whitelist for a replica set instance.
4. Apply for a public endpoint for a replica set instance.
5. Connect to a replica set instance. For more information, see Overview of replica set instance connections.
3 Create a replica set instance

This topic describes how to create a replica set 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 left-side navigation pane, click Replica Set Instances.

3. On the Replica Set Instances page, click Create Instance.

4. Click Subscription(Replica Set) or Pay-As-You-Go(Replica Set).

Note:

- Subscription: You must pay for the subscription 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.
5. Configure the instance. The following table describes related parameters.

<table>
<thead>
<tr>
<th>Section</th>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic Configuration</td>
<td>Region</td>
<td>The region where the replica set instance is deployed. Only instances in the same region can communicate with each other inside an internal network. After an instance is created, you cannot change its region. Exercise caution when you select the region. Instances in the same region (such as an <strong>ECS</strong> instance and an ApsaraDB for MongoDB instance) can be interconnected with each other through an internal network.</td>
</tr>
<tr>
<td>Zone</td>
<td></td>
<td>A <strong>zone</strong> 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 an internal network with the minimum network latency.</td>
</tr>
<tr>
<td>Database Version</td>
<td></td>
<td>The version of the database engine for the replica set instance. ApsaraDB for MongoDB supports 3.2, 3.4, and 4.0. We recommend that you select version 3.2 or later. For more information, see Version and storage engine.</td>
</tr>
<tr>
<td>Storage Engine</td>
<td></td>
<td>Select WiredTiger.</td>
</tr>
<tr>
<td>Section</td>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>Replication Factor</strong></td>
<td></td>
<td>Select the number of nodes for the replica set instance based on your business needs. For example, you can select more nodes for business scenarios with more reads than writes.</td>
</tr>
<tr>
<td><strong>Network Type</strong></td>
<td>Classic</td>
<td>Cloud services in a classic network are not isolated. You can configure security groups or whitelist policies to block unauthorized access to the cloud services.</td>
</tr>
<tr>
<td></td>
<td>VPC</td>
<td>A VPC is an isolated network with higher security and performance than a classic network.</td>
</tr>
<tr>
<td><strong>Specification</strong></td>
<td></td>
<td>The CPU and memory occupied by the instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• The maximum number of connections and maximum IOPS vary depending on different specifications. The maximum IOPS is measured for read and write operations separately, and the maximum sum of read and write operations can be twice the maximum IOPS. For more information, see #unique_27.</td>
</tr>
<tr>
<td><strong>Storage Space</strong></td>
<td></td>
<td>The storage space is exclusive to each node in the replica set instance.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• The storage space of a node stores your data, system, and log files.</td>
</tr>
</tbody>
</table>

**Note:**

- You must create a VPC in advance. For more information, see [Create a VPC](#).
- You can change the network type after you create an instance. For more information, see [Modify the instance network type](#).
- 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 [Configure a hybrid access solution to switch the network type of an ApsaraDB for MongoDB instance from Classic Network to VPC](#).
### Section: Set Password

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set Now</td>
<td>The password of the root user. You can set a password immediately or reset it during the running of the instance. For more information, see <em>Set a password</em>.</td>
</tr>
</tbody>
</table>
| Set Later | • 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. |

### Section: Purchase Quantity

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Duration</td>
<td>• Subscription: Select the duration and quantity for the subscription-based instance to be purchased. You can select one to nine months for the subscription period on a monthly basis, or one to three years for the subscription period on a yearly basis.</td>
</tr>
<tr>
<td>Quantity</td>
<td>• Pay-As-You-Go: Select the quantity for the Pay-As-You-Go instance to be purchased with the same configuration. You can select an integer in the range of 1 to 10.</td>
</tr>
</tbody>
</table>

6. Click Buy Now to go to the Confirm Order page.
7. On the Confirm Order page that appears, read and select ApsaraDB for MongoDB Agreement of Service and complete the payment as prompted.

### View the created instance

1. **Log on to the ApsaraDB for MongoDB console.**
2. **In the upper-left corner of the page, select the region where your instance resides.**
3. In the left-side navigation pane, click Replica Set Instances.

Troubleshoot if you cannot find the instance

<table>
<thead>
<tr>
<th>Possible cause</th>
<th>Troubleshooting</th>
</tr>
</thead>
<tbody>
<tr>
<td>You selected the wrong region in the console.</td>
<td>Select the region where the instance is deployed. For more information, see View created instances.</td>
</tr>
<tr>
<td>You opened the incorrect page.</td>
<td>In the left-side navigation pane, click Replica Set Instances. For more information, see View created instances.</td>
</tr>
<tr>
<td>The instance list in the ApsaraDB for MongoDB console was not updated or was updated before the instance is created.</td>
<td>Wait for several minutes and then update the instance list to check whether the instance is added to the list.</td>
</tr>
<tr>
<td>Resources are insufficient.</td>
<td>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 Orders page. After you confirm the refunded fees, you can try to create the instance in another zone. You can also submit a ticket.</td>
</tr>
</tbody>
</table>

What's next

After you create a replica set instance, you must configure a whitelist (for more information, see Configure a whitelist for a replica set 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 replica set instance).

For more information about instance connection methods and connection scenarios, see #unique_29.
4 Set a password for a replica set instance

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

Procedure

1. Log on to the ApsaraDB for MongoDB console.
2. In the upper-left corner of the page, select the region where your instance resides.
3. In the left-side navigation pane, click Replica Set 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.
This topic describes how to configure a whitelist for a replica set 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. In the left-side navigation pane, choose Data Security > Whitelist Setting.
2. Click the icon in the Operation 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.
Configure a whitelist for a replica set instance

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.

More operations

- #unique_30/unique_30_Connect_42_section_fwu_oit_4dc
- #unique_30/unique_30_Connect_42_section_1si_mlr_q72
Common connection scenarios

- #unique_31
- #unique_32
- #unique_33
- #unique_34

Result

After a whitelist is configured, the VPC connection address of the instance is displayed on the Basic Information and Database Connection pages.
6 Apply for a public endpoint for a replica set instance

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

Context

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

<table>
<thead>
<tr>
<th>Connection type</th>
<th>Description</th>
</tr>
</thead>
</table>
| Intranet Connection - VPC        | - A VPC is an isolated network with higher security and performance than a classic network.  
                                  | - By default, ApsaraDB for MongoDB provides endpoints on a VPC.                                                                                       |
| 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 Switch the network type of an ApsaraDB for MongoDB instance. |
| Public IP Connection             | - Connecting to a replica set instance over the Internet is risky. Therefore, ApsaraDB for MongoDB does not provide public endpoints.  
                                  | - 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. |

Procedure

1. Log on to the ApsaraDB for MongoDB console.
2. In the upper-left corner of the page, select the region where the target instance resides.
3. In the left-side navigation pane, click Replica Set Instances.
4. Find the target instance and click its ID.
5. In the left-side navigation pane, click Database Connection.
6. In the Public IP Connection section, click Apply for Public Connection String.

7. In the Apply for Public Connection String message that appears, click OK.

Note:
If you want to connect to a replica set instance by using a public endpoint, you must add the public IP address of your client to a whitelist of this instance. For more information, see Configure a whitelist for a replica set instance.

After the application is complete, the replica set instance generates new endpoints for both the primary and secondary nodes and the corresponding connection string URI. For more information, see Overview of replica set instance connections.
7 Connect to an instance

7.1 Overview of replica set instance connections

This topic provides an overview of replica set instance connections. ApsaraDB for MongoDB supports both connection strings and connection string URIs. You can use a connection string to connect to either the primary or secondary node, and use a connection string URI to connect to both of them. For high availability, we recommend that you use connection string URIs to connect your application to both primary and secondary nodes.

View connection addresses

1. Log on to the ApsaraDB for MongoDB console.
2. In the upper-left corner of the page, select the region where the target instance resides.
3. In the left-side navigation pane, click Replica Set Instances.
4. Find the target instance and click its ID.
5. In the left-side navigation pane, click Database Connection to view connection addresses.
## Introduction to connection addresses

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| **Address type** | • 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.  
                   • Intranet Connection - VPC: A VPC is an isolated network with higher security and performance than a classic network. By default, ApsaraDB for MongoDB provides endpoints on a VPC.  
                   • Public IP Connection: Connecting to a replica set instance over the Internet is risky. Therefore, ApsaraDB for MongoDB does not provide public endpoints. 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. For more information, see ![unique_36](#). |
| **Role**      | • Primary: the primary node in the replica set instance. If you connect to this node, you can perform read/write operations on the databases of the replica set instance.  
                   • Secondary: the secondary node in the replica set instance. If you connect to this node, you can only perform read operations on the databases of the replica set instance.  
                   **Note:** You can add secondary nodes to a replica set instance. For more information, see ![Change the number of nodes for a replica set instance](#). |
| **Connection string** | The connection string of a primary or secondary node is in the following format:  
                   `<host>:`:<port>  
                   • `<host>`: the endpoint you use to connect to the replica set instance  
                   • `<port>`: the port you use to connect to the replica set instance.  
                   **Note:** During regular tests, you can use a connection string to directly connect to a primary node. Note that a *failover* changes the roles of connected nodes, which affects read/write operations. |
## Connect to an instance

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Connection string URI</strong></td>
<td>A connection string URI is in the following format:</td>
</tr>
<tr>
<td></td>
<td><strong>mongodb://[username:password@]host1[:port1][,host2[:port2],...[,hostN[:portN]][/][database][? options]</strong></td>
</tr>
<tr>
<td></td>
<td>• <strong>mongodb://</strong>: the prefix, which indicates that the connection address is a connection string URI.</td>
</tr>
<tr>
<td></td>
<td>• <strong>username:password@</strong>: the username and password you use to log on to a database of the replica set instance. You must separate them with a colon (:).</td>
</tr>
<tr>
<td></td>
<td>• <strong>hostX:portX</strong>: the endpoint and port of a node in the replica set instance.</td>
</tr>
<tr>
<td></td>
<td>• <strong>/database</strong>: the name of the authentication database. It is the database where the database user is created.</td>
</tr>
<tr>
<td></td>
<td>• <strong>? options</strong>: additional connection options.</td>
</tr>
</tbody>
</table>

**Note:**
If your application is in a production environment, we recommend that you use a connection string URI to connect to the instance. This way, when a node fails, the read/write operations of your application are not affected as a result of the failover.

### Log on to a database of the replica set instance

1. **Obtain the connection addresses** and the following information:

   - The username you use 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 users 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 replica set instance*.

   - The name of the authentication database. It is the database where the database user is created. If the database username is root, enter admin.
2. Log on to the database.

- Connect to a replica set instance of ApsaraDB for MongoDB by using DMS
- Connect to a replica set instance of ApsaraDB for MongoDB by using the mongo shell
- Connect to a replica set instance of ApsaraDB for MongoDB by using code

Common connection scenarios

- #unique_31
- #unique_32
- #unique_33
- #unique_34
- Connect to a replica set instance to achieve read/write split and high availability

FAQ

- How to troubleshoot logon issues for the mongo shell
- #unique_44
- How to troubleshoot the high CPU usage of ApsaraDB for MongoDB
- How to query and limit the number of connections

7.2 Connect to a replica set 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 replica set 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 replica set 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.
Table 7-1: IP addresses of the DMS server

<table>
<thead>
<tr>
<th>Network type of ApsaraDB for MongoDB instance</th>
<th>IP address of the DMS server</th>
</tr>
</thead>
<tbody>
<tr>
<td>VPC</td>
<td>100.104.0.0/16</td>
</tr>
<tr>
<td>Classic network</td>
<td>120.55.177.0/24</td>
</tr>
<tr>
<td></td>
<td>121.43.18.0/24</td>
</tr>
<tr>
<td></td>
<td>101.37.74.0/24</td>
</tr>
<tr>
<td></td>
<td>10.153.176.0/24</td>
</tr>
<tr>
<td></td>
<td>10.137.42.0/24</td>
</tr>
<tr>
<td></td>
<td>11.193.54.0/24</td>
</tr>
<tr>
<td></td>
<td>10.152.163.0/24</td>
</tr>
</tbody>
</table>

Procedure

1. Log on to the ApsaraDB for MongoDB console.

2. In the upper-left corner of the page, select the region where the target instance resides.

3. In the left-side navigation pane, click Replica Set Instances.

4. Find the target instance and click its ID.

5. Click Logon and select Primary or Secondary in the upper-right corner of the Basic Information page. You are redirected to the DMS console.

Note:

- Primary: the primary node of the replica set instance. This node has read/write permissions on the database.
Secondary: the secondary node of the replica set instance. This node only has read permissions on the database.

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

![RDS Database Logon](image)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Network address:Port</strong></td>
<td>The internal connection string of the primary or secondary node of the ApsaraDB for MongoDB instance is automatically entered.</td>
</tr>
<tr>
<td><strong>Database Username</strong></td>
<td>Enter the database account of the ApsaraDB for MongoDB instance. The initial account is root.</td>
</tr>
<tr>
<td><strong>Database Name</strong></td>
<td>Enter the name of the database to which the account belongs.</td>
</tr>
<tr>
<td><strong>Password</strong></td>
<td>The password of the specified account.</td>
</tr>
</tbody>
</table>

**Note:**

- If Database Username is set to root, the database name is admin.
- 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 [Use DMS to manage ApsaraDB for MongoDB users](#).

- If you forget the password of the root account, you can reset the password by using the method specified in [Set a password](#).
7. Click Log On.

Common connection scenarios

- #unique_31
- #unique_32
- #unique_33
- #unique_34

FAQ

- How to troubleshoot logon issues for the mongo shell
- #unique_44
- Troubleshoot high CPU utilization of ApsaraDB for MongoDB
- How to query and limit the number of connections

7.3 Connect to a replica set instance by using the mongo shell

This topic describes how to connect to a replica set 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 in an ECS instance.

Prerequisites

- Mongo shell 3.0 or later is installed. For more information about the installation procedure, visit Install MongoDB at the official MongoDB website.
- The IP address of your client is added to a whitelist of the replica set instance.
  For more information, see Configure a whitelist for a replica set 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 region where the target instance resides.
3. In the left-side navigation pane, click Replica Set Instances.
4. Find the target instance and click its ID.
5. In the left-side navigation pane, click Database Connection to obtain the connection addresses of a node.

Note:
For more information about the connection addresses, see Introduction to connection addresses.

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

   - Single-node connection

   During regular tests, you can directly connect to a primary or secondary node. Note that a failover changes the roles of connected nodes, which affects read/write operations.

   Command format:

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

   Note:

   - `<host>`: the connection string of the primary or secondary node.

     ■ Primary node: If you connect to this node, you can perform read/write operations on the databases of the replica set instance.

     ■ Secondary node: If you connect to this node, you can only perform read operations on the databases of the replica set instance.
- `<username>`: the username you use to log on to a database of the replica set 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 users and grant permissions to them as needed. For more information, see #unique_38.

- `<database>`: the name of the authentication database. It is the database where the database user is created. If the database username is root, enter `admin`.

Example:

```bash
geno --host dds-bp**********.mongodb.rds.aliyuncs.com:3717 -u root -p --authenticationDatabase admin
```

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 replica set instance`.

**Note:**
The password you enter is not displayed.

- **HA connection (recommended):** You can use a connection string URI to connect to both the primary and secondary nodes of a replica set instance. This guarantees that your application is always connected to the primary node.
and the read/write operations of your application are not affected even if the roles of the primary and secondary nodes are switched.

Command format:

```
mongo "<ConnectionStringURI>"
```

Note:

- The connection string URI must be enclosed in a pair of double quotation marks (""").
- `<ConnectionStringURI>`: the Connection String URI of the replica set instance.

You must replace **** in the Connection String URI with the database password. For more information about how to set a database password, see Set a password for a replica set instance.

Common connection scenarios

- #unique_31

Note:

Before you connect to the replica set instance over the Internet, we recommend that you enable SSL encryption. For more information, see #unique_48.

- #unique_32
- #unique_33
FAQ

- How to troubleshoot logon issues for the mongo shell
- How to troubleshoot the high CPU usage of ApsaraDB for MongoDB
- How to query and limit the number of connections

7.4 Connection sample code for MongoDB drivers

Related links

- MongoDB Drivers
- Connection String URI Format

Note:
The connection sample code in this topic applies only when you use intranet addresses provided by Alibaba Cloud to connect to ApsaraDB for MongoDB.

Node.js

Related link: MongoDB Node.js Driver

1. Initialize a project.

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

2. Install the driver package and tool kit.

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

3. Obtain the information required to connect to an ApsaraDB for MongoDB instance.

   For more information, see Overview of replica set instance connections.

4. Use the following Node.js sample code.

```javascript
'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 the username and password used to log on to
    // ApsaraDB for MongoDB. The username in this sample code is used to
    // log on to 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.
            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 link:

MongoDB PHP Driver
1. Install the driver package and tool kit.

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

2. Obtain the information required to connect to an ApsaraDB for MongoDB instance.

   For more information, see Overview of replica set instance connections.

3. Use the following PHP sample code.

   ```php
   require 'vendor/autoload.php'; // include Composer goodies
   # Specify the 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 a MongoDB connection string URI 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;
   $client = new MongoDB\Client($demo_uri);
   $collection = $client->testDb->testColl;
   $result = $collection->insertOne(["name" => 'ApsaraDB for Mongodb', 'desc' => 'Hello, Mongodb']);
   echo "Inserted with Object ID '{$result->getInsertedId()}";"; echo "$result->getInsertedId()";";
   $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 the information required to connect to an ApsaraDB for MongoDB instance.

   For more information, see Overview of replica set instance connections.
2. Use the following Java sample code.

   • Maven configuration

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

• Java sample code

```java
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.MongoCredential;
import com.mongodb.MongoClientURI;
import com.mongodb.MongoCollection;
import com.mongodb.MongoDatabase;
import com.mongodb.ServerAddress;
import com.mongodb.client.MongoCursor;

public class Main {
  public static ServerAddress seed1 = new ServerAddress("demotest-1.mongodb.tbc3.newtest.rdstest.aliyun-inc.com", 27017);
  public static ServerAddress seed2 = new ServerAddress("demotest-2.mongodb.tbc3.newtest.rdstest.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<>();
    seedList.add(seed1);
    seedList.add(seed2);
    // Construct authentication information.
    List<MongoCredential> credentials = new ArrayList<>();
    credentials.add(MongoCredential.createScramSha1Credential(username, DEFAULT_DB, password.toCharArray()));
    // Construct operation options. Configure options other than requiredReplicaSetName based on your actual requirements. Default parameter settings can meet requirements in most scenarios.
    MongoClientOptions options = MongoClientOptions.builder().requiredReplicaSetName(ReplSetName)
      .sslEnabled(true)
      .socketTimeout(2000).connectionSPerHost(1).build();

    // Connect to a MongoDB instance.
    MongoClient mongoClient = MongoClient.createUncappedConnection(seedList.iterator(), credentials.iterator(), options);

    // Configure database and collection.
    MongoDatabase db = mongoClient.getDatabase(DEFAULT_DB);
    MongoCollection<Document> coll = db.getCollection(DEMO_COLL);

    // Perform operations on the collection.
    // Example: Insert a document
    Document doc = new Document("name", "example");
    coll.insertOne(doc);

    // Close the connection
    mongoClient.close();
  }
}
```
```java
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.
        MongoDatabase database = client.getDatabase(e(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("MESG", "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.
   
   ```bash
   pip install pymongo
   ```

2. Obtain the information required to connect to an ApsaraDB for MongoDB instance.
   
   For more information, see *Overview of replica set instance connections*.

3. Use the following Python sample code.
   
   ```python
   import uuid
   from pymongo import MongoClient
   
   # Specify two addresses for connecting 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 the username and password used to log on to ApsaraDB for MongoDB. The username in this sample code is used to log on to the admin database.
   client.admin.authenticate(username, password)
   
   # Use the collection:testColl of the test database as an example.
   demo_name = 'python-' + str(uuid.uuid1())
   print 'demo_name:', demo_name
   doc = dict(DEMO=demo_name, MESG="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
   ```

C# Related links: *MongoDB C# Driver*.

1. Obtain the information required to connect to an ApsaraDB for MongoDB instance. For more information, see *Overview of replica set instance connections*.

2. Install the driver package.
   
   ```bash
   mongocsharpdriver.dll
   ```

3. C# Demo Code.
   
   ```csharp
   using MongoDB.Driver;
   using System;
   using System.Collections.Generic;
   namespace Aliyun
   {
   ```
```csharp
class Program
{
    static void Main(string[] args)
    {
        //Mongo Instance Information
        const string host1 = "dds-t4nffb7491eb87541.mongodb.singapore.rds.aliyuncs.com";
        const int port1 = 3717;
        const string host2 = "dds-t4nffb7491eb87542.mongodb.singapore.rds.aliyuncs.com";
        const int port2 = 3717;
        const string replicaSetName = "mgset-300374302";
        const string admin = "admin";
        const string userName = "root";
        const string passwd = "Lsx111222";
        try
        {
            Console.WriteLine("connecting...");
            MongoClientSettings settings = new MongoClientSettings();
            List<MongoServerAddress> servers = new List<MongoServerAddress>
                { servers.Add(new MongoServerAddress(host1, port1));
                servers.Add(new MongoServerAddress(host2, port2));
                settings.Servers = servers;
                //set ReplicaSetName
                settings.ReplicaSetName = replicaSetName;
                //set ConnectTimeout to 3
                settings.ConnectTimeout = new TimeSpan(0, 0, 0, 3, 0);
                MongoCredential credentials = MongoCredential.
                    CreateCredential(admin, userName, passwd);
                settings.Credential = credentials;
                MongoClient client = new MongoClient(settings);
                var server = client.GetServer();
                MongoDatabase database = server.GetDatabase("test");
                var collection = database.GetCollection<User>("test_collection");
                User user = new User();
                user.id = "1";
                user.name = "mongo_test";
                user.sex = "Female";
                //insert data user
                collection.Insert(user);
                //obtain a data
                User result = collection.FindOne();
                Console.WriteLine("id:" + result.id + " name:" +
                result.name + " sex:" + result.sex);
                Console.WriteLine("connection successful........");
            }
            catch (Exception e)
            {
                Console.WriteLine("connection failed:"+e.Message);
            }
        }
    }
}

class User
{
    public string id { set; get; }
    public string name { set; get; }
    public string sex { set; get; }
}
8 Migrate data

8.1 Migrate user-created MongoDB databases to Alibaba Cloud by using DTS

This topic describes how to migrate user-created MongoDB databases to a replica set instance of ApsaraDB for MongoDB by using Data Transmission Service (DTS). DTS allows you to fully and incrementally migrate data without interruptions to your applications.

To avoid service disruption, we recommend that you use DTS to migrate user-created MongoDB databases to Alibaba Cloud. You can also use the built-in commands of MongoDB in this situation. For more information, see Migrate user-created MongoDB databases to Alibaba Cloud by using the built-in commands of MongoDB.

For more information about data migration and synchronization solutions, see "unique_50".

Prerequisites

- The service port of the user-created MongoDB databases is accessible over the Internet.
- The version of the user-created MongoDB databases is 3.0, 3.2, 3.4, 3.6, or 4.0.
- The storage capacity of the destination replica set instance is greater than the occupied storage space of the user-created MongoDB databases.

Precautions

- We recommend that you migrate the user-created MongoDB databases during off-peak hours to avoid business interruptions.
- If the source user-created MongoDB databases and the destination replica set instance run different database versions or storage engines, ensure that there are no compatibility issues between them before you start migration. For more information about the database versions and storage engines supported by ApsaraDB for MongoDB, see "unique_4".
Billing

<table>
<thead>
<tr>
<th>Migration type</th>
<th>Link configuration fee</th>
<th>Internet traffic fee</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full data migration</td>
<td>Free of charge.</td>
<td>Charged only when data is migrated from Alibaba Cloud over the Internet.</td>
</tr>
<tr>
<td>Incremental data migration</td>
<td>Charged. For more information, see #unique_51.</td>
<td>For more information, see #unique_51.</td>
</tr>
</tbody>
</table>

Migration types

- Full data migration: All data of the migration objects is migrated from a source database to a destination database.

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

- Incremental data migration: Updated data of the migration objects is synchronized from a source database to a destination database.

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

Required database account permissions

<table>
<thead>
<tr>
<th>Data source</th>
<th>Full data migration</th>
<th>Incremental data migration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source user-created MongoDB database</td>
<td>Read permissions on the source database</td>
<td>Read permissions on the source database, admin database, and local database</td>
</tr>
<tr>
<td>Destination replica set instance of ApsaraDB for MongoDB</td>
<td>Read/write permissions on the destination database</td>
<td>Read/write permissions on the destination database</td>
</tr>
</tbody>
</table>

For more information about how to create and authorize a database account:

- For a replica set instance of ApsaraDB for MongoDB, see Manage MongoDB users through DMS.
• For user-created MongoDB databases, visit *Create User in MongoDB*.

**Procedure**

1. Log on to the *DTS console*.
2. In the left-side navigation pane, click Data Migration.
3. In the Migration Tasks section, select the region where the destination replica set instance resides.

4. In the upper-right corner, click Create Migration Task.
5. Configure the source and destination databases.

<table>
<thead>
<tr>
<th>Section</th>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Task Name</td>
<td>N/A</td>
<td>DTS automatically generates a task name. We recommend that you specify your own task name that helps identify the task. Task names do not need to be unique.</td>
</tr>
<tr>
<td>Section</td>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>-----------------</td>
<td>---------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Source Database</td>
<td>Instance Type</td>
<td>Select User-Created Database with Public IP Address.</td>
</tr>
<tr>
<td></td>
<td>Instance Region</td>
<td>If you set Instance Type to User-Created Database with Public IP Address, the system automatically specifies Instance Region.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> If the source database has a whitelist, you must click Get IP Address Segment of DTS next to Instance Region to obtain the Classless Inter-Domain Routing (CIDR) block of DTS servers, and add it to the whitelist.</td>
</tr>
<tr>
<td></td>
<td>Database Type</td>
<td>Select MongoDB.</td>
</tr>
<tr>
<td></td>
<td>Hostname or IP Address</td>
<td>Enter a connection address of the source database. For this example, enter a public connection address.</td>
</tr>
<tr>
<td></td>
<td>Port Number</td>
<td>Enter the service port of the source database.</td>
</tr>
<tr>
<td></td>
<td>Database Name</td>
<td>Enter the name of the authentication database. It is the database where the database account is created.</td>
</tr>
<tr>
<td></td>
<td>Database Account</td>
<td>Enter the username of the database account you use to manage the source database. For more information about the account permission requirements, see Required database account permissions.</td>
</tr>
<tr>
<td></td>
<td>Database Password</td>
<td>Enter the password of the database account.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> After you specify the source database information, click Test Connectivity next to Database Password to check whether the information is correct. If the information is correct, the Passed message is displayed. If the information is incorrect, the Failed message is displayed, and you must click Check next to the Failed message to modify the information as prompted.</td>
</tr>
<tr>
<td>Destination Database</td>
<td>Instance Type</td>
<td>Select MongoDB Instance.</td>
</tr>
<tr>
<td></td>
<td>Instance Region</td>
<td>Select the region where the destination replica set instance resides.</td>
</tr>
<tr>
<td>Section</td>
<td>Parameter</td>
<td>Description</td>
</tr>
<tr>
<td>--------------</td>
<td>----------------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td>MongoDB Instance ID</td>
<td>Select the ID of the destination replica set instance.</td>
</tr>
<tr>
<td></td>
<td>Database Name</td>
<td>Enter the name of the authentication database. It is the database where the database account is created.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> If the database account is root, enter admin.</td>
</tr>
<tr>
<td></td>
<td>Database Account</td>
<td>Enter the username of the database account you use to manage the destination database. For more information about the account permission requirements, see Required database account permissions.</td>
</tr>
<tr>
<td></td>
<td>Database Password</td>
<td>Enter the password of the database account.</td>
</tr>
<tr>
<td></td>
<td></td>
<td><strong>Note:</strong> After you specify the destination database information, click Test Connectivity next to Database Password to check whether the information is correct. If the information is correct, the Passed message is displayed. If the information is incorrect, the Failed message is displayed, and you must click Check next to the Failed message to modify the information as prompted.</td>
</tr>
</tbody>
</table>

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

**Note:**
The IP addresses of DTS servers are automatically added to a whitelist of the destination replica set instance. This ensures that the DTS servers can connect to this instance. After the migration is complete, you can remove the IP addresses from the whitelist. For more information, see Configure a whitelist for a replica set instance.
7. **Configure migration types and migration objects.**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Migration Types</td>
<td>- If you want to migrate all data, select Full Data Migration.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> To ensure data consistency, do not write data to the user-created MongoDB database while full data migration is in progress.</td>
</tr>
<tr>
<td></td>
<td>- If you want to migrate data without interruptions to your business, select both Full Data Migration and Incremental Data Migration.</td>
</tr>
<tr>
<td>Available</td>
<td>- In the Available section, select the objects you want to migrate and then click the <strong>»</strong> icon to move them to the Selected section.</td>
</tr>
</tbody>
</table>
|                    | **Note:**  
|                    |   - Data in the admin database cannot be migrated even if this database is selected.                                                                                                                       |
|                    |   - The config database is an internal database. Do not migrate data in this database unless otherwise specified.                                                                                           |
|                    |   - A migration object can be a database, collection, or function.                                                                                                                                          |
|                    |   - By default, the name of an object remains unchanged after migration. If you want a different object name after migration, use the object name mapping feature provided by DTS. For more information, see *Object name mapping*. |
8. In the lower-right corner, click Precheck.

Note:

- A precheck is performed before the migration task starts. The migration task starts only after the precheck succeeds.
- If the precheck fails, click the icon for each failed check item to view their details. Perform a precheck again after the failures are fixed.

9. After the precheck succeeds, click Next.


11. Click Buy and Start to start the migration task.

- Full data migration
  
  Do not manually end a migration task. If you do so, the system may fail to migrate all data of the database. Wait until the migration task is complete.

- Incremental data migration
  
  An incremental data migration task does not automatically end. You must manually end the task.

Note:

Select an appropriate time to manually end a migration task. For example, you can end the migration task during off-peak hours or before you switch over your business to the destination replica set instance.

a. When the task progress bar displays Incremental Data Migration and The migration task is not delayed, stop writing data to the source database for a few minutes. Wait until the progress bar displays the delay time of the incremental data migration next to Incremental Data Migration.

b. After the status of Incremental Data Migration changes to The migration task is not delayed, manually end the migration task.
12. Switch over your business to the destination replica set instance.

References

Overview of replica set instance connections

8.2 Migrate user-created MongoDB databases to Alibaba Cloud by using the built-in commands of MongoDB

This topic describes how to migrate user-created MongoDB databases to Alibaba Cloud by using mongodump and mongorestore, which are both built in MongoDB for 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 a replica set instance of ApsaraDB for MongoDB.

To avoid service disruption, we recommend that you use DTS to migrate user-created MongoDB databases to Alibaba Cloud. For more information, see Migrate user-created MongoDB databases to Alibaba Cloud by using DTS.

For more information about data migration and synchronization solutions, see #unique_50.

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 official MongoDB 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 replica set instance is greater than the occupied storage space of the user-created MongoDB databases. If the storage capacity is insufficient, you can upgrade the instance. For more information, see #unique_52.
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 used 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 in 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 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 you use to log on to a user-created MongoDB database.
   - `<database>`: the name of the authentication database. It is the database where the database user is created.

   **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.

   **Note:**
   The password you enter is not displayed.
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 Migrate data to the destination replica set instance

1. Obtain the public or internal connection string of the primary node in the destination replica set instance. For more information, see Overview of replica set instance connections.

   **Note:**
   You must apply for a public endpoint manually. For more information, see Apply for a public endpoint for a replica set instance.

2. Add the IP address of the server where the user-created MongoDB databases reside to a whitelist of the destination replica set instance. For more information, see Configure a whitelist for a replica set instance.

   **Note:**
   - If you want to connect to a replica set instance over an 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 replica set instance.
   - If you want to connect to a replica set 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 replica set 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 replica set instance:

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

   **Note:**
   - `<Primary_host>`: the connection string of the primary node in the destination replica set instance.
   - `<username>`: the username you use to log on to a database of the destination replica set instance. The initial username is root.
ApsaraDB for MongoDB

Quick Start for Replica Set / Migrate data

• `<database>`: the name of the authentication database. It is the database where the database user is created. If the database username is root, enter admin.

• `<Backup directory>`: the directory where the backup files are stored. The default value is `dump`.

Example:

```
mongorestore --host dds-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 you enter is not displayed.
- If you forget the password of the root user, you can reset it. For more information, see `Set a password for a replica set instance`.

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