

# Distributed Database Middleware

## FAQs

**Issue** 01  
**Date** 2024-07-30



**Copyright © Huawei Technologies Co., Ltd. 2024. All rights reserved.**

No part of this document may be reproduced or transmitted in any form or by any means without prior written consent of Huawei Technologies Co., Ltd.

## **Trademarks and Permissions**



HUAWEI and other Huawei trademarks are trademarks of Huawei Technologies Co., Ltd.

All other trademarks and trade names mentioned in this document are the property of their respective holders.

## **Notice**

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

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

# Security Declaration

## Vulnerability

Huawei's regulations on product vulnerability management are subject to the *Vul. Response Process*. For details about this process, visit the following web page:

<https://www.huawei.com/en/psirt/vul-response-process>

For vulnerability information, enterprise customers can visit the following web page:

<https://securitybulletin.huawei.com/enterprise/en/security-advisory>

---

# Contents

---

<b>1 General Questions.....</b>	<b>1</b>
1.1 What High-Reliability Mechanisms Does DDM Provide?.....	1
1.2 How Do I Select and Configure a Security Group?.....	1
1.3 What Do I Do If My Database Time Is 13 or 14 Hours Later than Beijing Time?.....	3
1.4 Can Data Nodes Associated with a DDM Instance Share Data?.....	3
1.5 What Data Nodes Can Be Associated with a DDM Instance?.....	3
<b>2 DDM Usage.....</b>	<b>5</b>
2.1 How Does DDM Perform Sharding?.....	5
2.2 What Do I Do If I Fail to Connect to a DDM Instance Using the JDBC Driver?.....	6
2.3 Why It Takes So Long Time to Export Data from MySQL Using mysqldump?.....	7
2.4 What Do I Do If a Duplicate Primary Key Error Occurs When Data Is Imported into DDM?.....	7
2.5 What Should I Do If an Error Message Is Returned When I Specify an Auto-Increment Primary Key During Migration?.....	7
2.6 What Do I Do If an Error Is Reported When Parameter Configuration Does Not Time Out?.....	8
2.7 Which Should I Delete First, a Schema or its Associated RDS Instances?.....	8
2.8 Can I Manually Delete Databases and Accounts Remained in Data Nodes After a Schema Is Deleted?.....	8
<b>3 SQL Syntax.....</b>	<b>9</b>
3.1 Does DDM Support Distributed JOINS?.....	9
3.2 How Do I Optimize SQL Statements?.....	9
3.3 Does DDM Support Forced Conversion of Data Types?.....	9
3.4 What Should I Do If an Error Is Reported When Multiple Data Records Are Inserted into Batches Using the INSERT Statement?.....	10
<b>4 RDS-related Questions.....</b>	<b>11</b>
4.1 Is the Name of a Database Table Case-Sensitive?.....	11
4.2 What Risky Operations on RDS for MySQL Will Affect DDM?.....	11
4.3 How Do I Handle Data with Duplicate Primary Keys in a Table?.....	13
4.4 How Can I Query RDS for MySQL Information by Running Command <b>show full innodb status</b> ?.....	14
4.5 What Should I Pay Attention to When Selecting RDS for MySQL Instance Specifications?.....	14
<b>5 Connection Management.....</b>	<b>16</b>
5.1 How Can I Handle Garbled Characters Generated When I Connect a MySQL Instance to a DDM Instance?.....	16

---

**6 Resource Freezing, Release, Deletion, and Unsubscription.....17**

# 1 General Questions

---

## 1.1 What High-Reliability Mechanisms Does DDM Provide?

### Protection of Data Integrity

DDM instance faults do not affect data integrity.

- Service data is stored in shards of data nodes, but not on DDM.
- Configuration information of schemas and logical tables is stored in DDM databases. Primary and standby DDM databases are highly available.

### High Availability

DDM is deployed using multiple stateless nodes in cluster mode and provides services through the IP address bound to your load balancer.

- If one DDM node becomes faulty, an error is returned for connections established on the node, without affecting the DDM cluster. The faulty node is generally deleted from the cluster within 5 seconds.
- If a data node becomes faulty, services can be restored within 30 seconds after the data node is recovered.

## 1.2 How Do I Select and Configure a Security Group?

DDM uses VPCs and security groups to ensure security of your instances. The following provides guidance for you on how to correctly configure a security group.

### Intra-VPC Access to DDM Instances

Access to a DDM instance includes access to the DDM instance from the ECS where a client is located and access to its associated data nodes.

The ECS, DDM instance, and data nodes must be in the same VPC. In addition, correct rules should be configured for their security groups to allow network access.

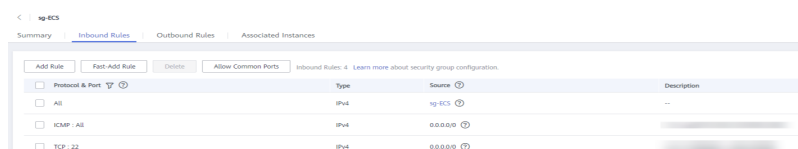
1. Using the same security group is recommended for the ECS, DDM instance, and data nodes. After a security group is created, network access in the group is not restricted by default.
2. If different security groups are configured, you may need to refer to the following configurations:

**NOTE**

- Assume that the ECS, DDM instance, and RDS for MySQL instance are configured with security groups **sg-ECS**, **sg-DDM**, and **sg-RDS**, respectively.
- Assume that the service port of the DDM instance is **5066** and that of the RDS for MySQL instance is **3306**.
- The remote end should be a security group or an IP address.

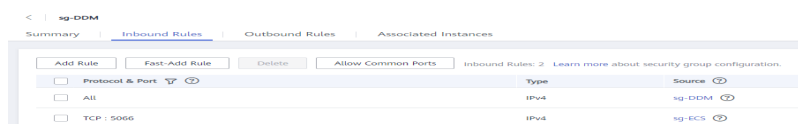
Add the rules described in the following figure to the security group of the ECS to ensure that your client can access the DDM instance.

**Figure 1-1** ECS security group rules

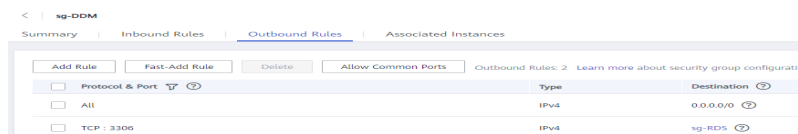


Add the rules shown in the following figure to the security group of the ECS where your DDM instance is located so that your DDM instance can access associated data nodes and can be accessed by your client.

**Figure 1-2** Configuring security group inbound rules for your DDM instance

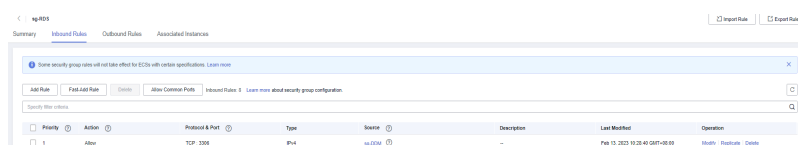


**Figure 1-3** Configuring security group outbound rules for your DDM instance

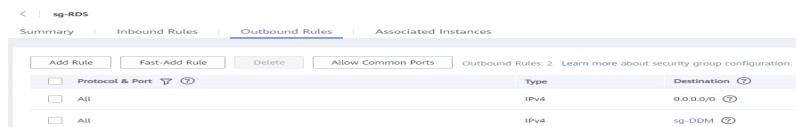


Add the rules shown in the following figure to the security group of the ECS where the data node is located so that your DDM instance can access the node.

**Figure 1-4** Configuring security group inbound rules for the associated RDS instance



**Figure 1-5** Configuring security group outbound rules for the associated RDS instance



## 1.3 What Do I Do If My Database Time Is 13 or 14 Hours Later than Beijing Time?

### Symptom

If GMT+8:00 is used in your database, your database time is 13 or 14 hours later than Beijing time when you connect to your DDM instance using JDBC.

### Possible Cause

If you connect to a DDM instance using JDBC, JDBC queries the database time zone of the instance and DDM returns China Standard Time UTC+08:00 to JDBC.

CST has the following meanings:

- Central Standard Time (USA) UTC-06:00
- Central Standard Time (Australia) UTC+09:30
- China Standard Time UTC+08:00
- Cuba Standard Time UTC-04:00

The JDBC driver parses CST as Central Standard Time (USA) UTC-06:00, which is 13 or 14 hours later than Beijing time.

### Solution

Add time zone configurations in the JDBC address:

```
jdbc:mysql://xxx:3306/database_name?serverTimezone=Asia/Shanghai&useUnicode=true&characterEncoding=utf8
```

## 1.4 Can Data Nodes Associated with a DDM Instance Share Data?

No. Different data nodes associated with a DDM instance are independent of each other and cannot share data.

## 1.5 What Data Nodes Can Be Associated with a DDM Instance?

Any data nodes can be associated with a DDM instance as long as they are:



- Running normally
- In the same VPC as the DDM instance
- Not in use by other DDM instances

# 2 DDM Usage

---

## 2.1 How Does DDM Perform Sharding?

Distributed databases use shard-based storage, which removes capacity bottlenecks of single-node databases caused by a large amount of data in one unsharded table. Therefore, when creating a schema and logical tables, you need to consider your actual conditions and determine whether to create sharded tables and which sharding rule should be used.

### NOTE

Avoid cross-shard JOIN operations on the data that is stored in different shards, to ensure optimal performance and resource availability.

- Whether logical tables are sharded

DDM supports three types of logical tables: broadcast, sharded, and unsharded. You can select the most appropriate type based on your specific needs. For details, see [Creating a Table](#).

  - Unsharded: One physical table is created and stores data only in the first shard.
  - Broadcast: The same physical table is created in each shard and stores the same data.
  - Sharded: The same physical table is created in each shard, and data is distributed to all shards based on a certain sharding rule.
- Sharding rule of logical tables

The selection of a sharding key is important for each logical table. Selecting the sharding key based on your needs is recommended. If an entity relationship exists between different logical tables, select the same field as the sharding key to avoid cross-shard JOIN.

Pay attention to the following suggestions before determining whether to use sharding:

- Do not shard tables that each have less than 10 million data records.
- Shard the tables that each have more than 10 million data records. Storing data in different sharded tables removes performance bottlenecks caused by a

large amount of data in one table, while also improving concurrency capability. Select an appropriate sharding key in advance.

- Avoid across-table JOIN operations during service reading and cross-shard operations for any individual transaction.
- Include the sharding key into query conditions to avoid scanning table shards in all sharded tables.

## 2.2 What Do I Do If I Fail to Connect to a DDM Instance Using the JDBC Driver?

When you access a DDM instance using the MySQL driver (JDBC) in load balancing mode, an infinite loop may occur during connection switchover, resulting in stack overflow.

### Fault Locating

1. Query the application logs and locate the fault cause.

For example, the following logs show that the fault is caused by stack overflow.

```
Caused by: java.lang.StackOverflowError
  at java.nio.HeapByteBuffer.<init>(HeapByteBuffer.java:57)
  at java.nio.ByteBuffer.allocate(ByteBuffer.java:335)
  at java.nio.charset.CharsetEncoder.encode(CharsetEncoder.java:795)
  at java.nio.charset.Charset.encode(Charset.java:843)
  at com.mysql.jdbc.StringUtils.getBytes(StringUtils.java:2362)
  at com.mysql.jdbc.StringUtils.getBytes(StringUtils.java:2344)
  at com.mysql.jdbc.StringUtils.getBytes(StringUtils.java:568)
  at com.mysql.jdbc.StringUtils.getBytes(StringUtils.java:626)
  at com.mysql.jdbc.Buffer.writeStringNotNull(Buffer.java:670)
  at com.mysql.jdbc.MySqlIO.sqlQueryDirect(MySqlIO.java:2636)
```

2. Analyze the overflow source.

For example, the following logs show that the overflow results from an infinite loop inside the driver.

```
at
com.mysql.jdbc.LoadBalancedConnectionProxy.pickNewConnection(LoadBalancedConnectionProxy.java:
344)
at
com.mysql.jdbc.LoadBalancedAutoCommitInterceptor.postProcess(LoadBalancedAutoCommitIntercepto
r.java:104)
at com.mysql.jdbc.MySqlIO.invokeStatementInterceptorsPost(MySqlIO.java:2885)
at com.mysql.jdbc.MySqlIO.sqlQueryDirect(MySqlIO.java:2808)
at com.mysql.jdbc.ConnectionImpl.execSQL(ConnectionImpl.java:2483)
at com.mysql.jdbc.ConnectionImpl.setReadOnlyInternal(ConnectionImpl.java:4961)
at com.mysql.jdbc.ConnectionImpl.setReadOnly(ConnectionImpl.java:4954)
at com.mysql.jdbc.MultiHostConnectionProxy.syncSessionState(MultiHostConnectionProxy.java:381)
at com.mysql.jdbc.MultiHostConnectionProxy.syncSessionState(MultiHostConnectionProxy.java:366)
at
com.mysql.jdbc.LoadBalancedConnectionProxy.pickNewConnection(LoadBalancedConnectionProxy.java:
344)
```

3. Query the MySQL version, which is 5.1.44.

According to the source code of the version, when a connection is obtained, **LoadBalance** updates the connection based on the load balancing policy and copies the configurations of the old connection to the new connection. If **AutoCommit** is **true** for the new connection, parameters of the new connection are inconsistent with those of the old connection, and **loadBalanceAutoCommitStatementThreshold** is not configured, an infinite

loop occurs. The connection update function calls the parameter synchronization function, and the parameter synchronization function calls the connection update function at the same time, resulting in stack overflow.

## Solution

Add the **loadBalanceAutoCommitStatementThreshold=5&retriesAllDown=10** parameter to the URL for connecting to the DDM instance.

```
//Connection example when load balancing is used
//jdbc:mysql:loadbalance://ip1:port1,ip2:port2..ipN:portN/{db_name}
String url = "jdbc:mysql:loadbalance://192.168.0.200:5066,192.168.0.201:5066/db_5133?
loadBalanceAutoCommitStatementThreshold=5&retriesAllDown=10";
```

- **loadBalanceAutoCommitStatementThreshold** indicates the number of statements executed before a reconnection.

If **loadBalanceAutoCommitStatementThreshold** is set to **5**, a reconnection is initiated after five SQL statements (queries or updates) are executed. A value of **0** indicates a sticky connection, and no reconnection is required. When automatic submission is disabled (**autocommit** is set to **false**), the system waits for the transaction to complete and then determines whether to initiate a reconnection.

## 2.3 Why It Takes So Long Time to Export Data from MySQL Using mysqldump?

The version of the mysqldump client may be inconsistent with that of the supported MySQL server, so exporting data from MySQL is slow.

Using the same version of the mysqldump client and MySQL server is recommended.

## 2.4 What Do I Do If a Duplicate Primary Key Error Occurs When Data Is Imported into DDM?

When you create a table in DDM, set the start value for automatic increment and ensure that the start value is greater than the maximum auto-increment value of imported data.

## 2.5 What Should I Do If an Error Message Is Returned When I Specify an Auto-Increment Primary Key During Migration?

Execute the following SQL statement to modify the start value of the auto-increment primary key so that the value is greater than the maximum value of primary keys in existing tables:

```
ALTER SEQUENCE Database name.SEQ name START WITH New start value
```

## 2.6 What Do I Do If an Error Is Reported When Parameter Configuration Does Not Time Out?

Adjust the **SocketTimeOut** value or leave this parameter blank. The default value is **0**, indicating that the client is not disconnected.

## 2.7 Which Should I Delete First, a Schema or its Associated RDS Instances?

After an RDS instance is associated with your schema, you cannot delete the instance directly. To delete it, you have to delete the schema first and then delete the instance.

## 2.8 Can I Manually Delete Databases and Accounts Remained in Data Nodes After a Schema Is Deleted?

You can delete DDM instances or accounts if they are no longer needed.

For details, see [Deleting a DDM Instance](#) and [Deleting an Account](#).

# 3 SQL Syntax

---

## 3.1 Does DDM Support Distributed JOINS?

Yes. DDM supports distributed JOINS.

- Redundant fields are added during table design.
- Cross-shard JOIN is implemented by using broadcast tables, ER shards, and ShareJoin.
- Currently, DDM does not allow cross-schema update or deletion of multiple tables.

## 3.2 How Do I Optimize SQL Statements?

- You are advised to use INNER instead of LEFT JOIN or RIGHT JOIN.
- When LEFT JOIN or RIGHT JOIN is used, ON is preferentially executed, and WHERE is executed at the end. Therefore, when using LEFT JOIN or RIGHT JOIN, ensure that the conditions are judged in the ON statement to reduce the execution of WHERE.
- When possible, use JOIN instead of subqueries to avoid full scanning of large tables.

## 3.3 Does DDM Support Forced Conversion of Data Types?

Data type conversion is an advanced function. DDM will be gradually upgraded to be compatible with more SQL syntax. If necessary, submit a service ticket for processing.

## **3.4 What Should I Do If an Error Is Reported When Multiple Data Records Are Inserted into Batches Using the INSERT Statement?**

### **Solution**

Split an INSERT statement into multiple small statements.

# 4 RDS-related Questions

## 4.1 Is the Name of a Database Table Case-Sensitive?

DDM is case-insensitive to database names, table names, column names, and column values by default.

## 4.2 What Risky Operations on RDS for MySQL Will Affect DDM?

[Table 4-1](#) lists risky operations on RDS for MySQL.

**Table 4-1** Risky operations on RDS for MySQL

Operation Type	Operation	Impact of the Operation
Operations on the RDS for MySQL console	Deleting an RDS for MySQL instance	After an RDS for MySQL instance is deleted, all schemas and logical tables of the DDM instance associated with the RDS instance become unavailable.
	Performing the primary/standby switchover of an RDS for MySQL instance	<p>RDS for MySQL may be intermittently interrupted during the primary/standby switchover. In addition, a small amount of data may be lost in case of long delay in primary/standby synchronization.</p> <ul style="list-style-type: none"> <li>• Creating schemas or logical tables is not allowed on DDM during the primary/standby switchover of the RDS for MySQL instance.</li> <li>• After a primary/standby switchover of an RDS for MySQL instance, the RDS instance ID remains unchanged in DDM.</li> </ul>



Operation Type	Operation	Impact of the Operation
	Restarting an RDS for MySQL instance	The restart of an RDS for MySQL instance makes itself unavailable and will also affect the associated DDM instance.
	Resetting a password	After the RDS for MySQL instance password is reset, enter the new password on DDM when you create a schema.
	Modifying a parameter template	<p>The following parameters are set to fixed values. If their values are modified, DDM will not function properly.</p> <ul style="list-style-type: none"> <li>• <b>lower_case_table_names</b>: Set this parameter to <b>1</b>, indicating that data table names and sequence names are case-insensitive.</li> <li>• <b>local_infile</b>: Set this parameter to <b>ON</b> in scale-out scenarios.</li> </ul>
	Modifying a security group	Your DDM instance cannot connect to associated RDS for MySQL instances.
	Modifying a VPC	The DDM instance and RDS for MySQL instance cannot communicate with each other if they are in different VPCs.
	Restoring data	Restoring data may damage data integrity.
Operations through an RDS for MySQL client	Deleting a physical database created on DDM	After a physical database is deleted, the original data will be lost and new data cannot be written into the database.
	Deleting an account created on DDM	After an account is deleted, logical tables cannot be created on DDM.
	Deleting a physical table created on DDM	After a physical table is deleted, data stored on DDM will be lost. The corresponding logical table becomes unavailable on DDM.

Operation Type	Operation	Impact of the Operation
	Modifying the name of a physical table created on DDM	DDM cannot obtain data of the corresponding logical table, and the logical table becomes unavailable on DDM.
	Changing a record	Changing a record in a broadcast table will affect the data consistency of shards.
	Modifying a whitelist	A DDM instance cannot access the RDS for MySQL instance if it is not in the RDS instance whitelist.

## 4.3 How Do I Handle Data with Duplicate Primary Keys in a Table?

### Scenario

If there is already a primary key whose data type is a boundary value, in your DDM instance, duplicate primary keys will be reported when you insert a data record that is beyond the data range of the primary key.

### Procedure

- Step 1** Log in to the management console.
- Step 2** On the **Instances** page, locate the RDS for MySQL instance associated with your DDM instance and click the name of the RDS instance.
- Step 3** On the **Basic Information** page, choose **Parameters** in the left pane.
- Step 4** Click the **Parameters** tab and enter **sql\_mode** in the text box. Then click the expanding button in the **Value** column, select **STRICT\_ALL\_TABLES** or **STRICT\_TRANS\_TABLES**, and click **Save**.

**NOTE**

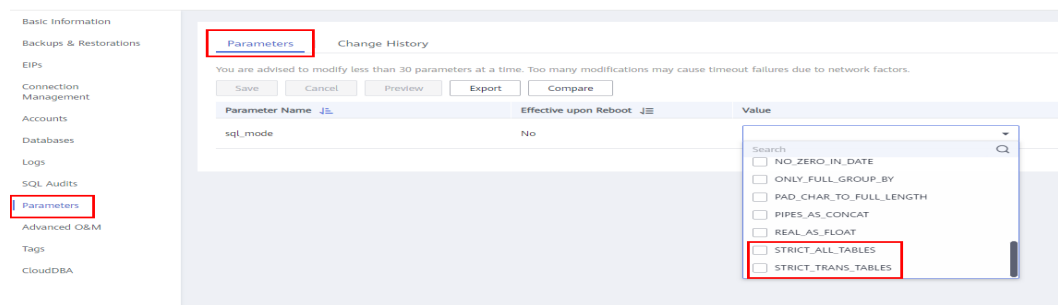
**STRICT\_ALL\_TABLES** and **STRICT\_TRANS\_TABLES** are both strict modes. The strict mode controls how MySQL handles invalid or missing values.

- An invalid value might have the wrong data type for the column, or might be out of range.
- A value is missing when a new row to be inserted does not contain a value for a non-NULL column that has no explicit DEFAULT clause in its definition.
- If the DDM instance version is earlier than 2.4.1.3, do not set **sql\_mode** to **ANSI\_QUOTES**. If you set it to **ANSI\_QUOTES**, double quotation marks used for each string will be translated into an identifier during SQL statement execution, making the string invalid.

For example, **logic** in **select \* from test where tb = "logic"** cannot be parsed correctly.

For more information about SQL modes, see [Server SQL Modes](#).

**Figure 4-1** Modifying instance parameters



**Step 5** On the **Instances** page, restart the DDM instance.

----End

## 4.4 How Can I Query RDS for MySQL Information by Running Command show full innodb status?

After you connect to a DDM instance through the MySQL client, you can run command **show full innodb status** to query information about the associated RDS for MySQL instances. The following information can be queried:

- Current time and duration since the last output.
- Status of the master thread.
- SEMAPHORES including event counts and available waiting threads when there is high-concurrency workload. You can use the information to locate performance bottlenecks if any.

## 4.5 What Should I Pay Attention to When Selecting RDS for MySQL Instance Specifications?

Specifications of associated RDS for MySQL instances should be greater than that of the DDM instance. Otherwise the performance will be affected.

Example:

Question: If a DDM instance has 8 vCPUs and 16 GB memory, which specification should I select for its associated RDS for MySQL instances? Should I use a primary/standby or single-node architecture?

Answer: According to best practices, the configuration ratio of DDM instances to RDS for MySQL instances should remain 1:2. If a DDM instance has 8 vCPUs and 16 GB memory, select at least two RDS for MySQL instances with 8 vCPUs and 16 GB memory. The number of required RDS for MySQL instances depends on your service requirements.

To run production workloads, select primary/standby RDS for MySQL instances to ensure high availability.

# 5 Connection Management

---

## 5.1 How Can I Handle Garbled Characters Generated When I Connect a MySQL Instance to a DDM Instance?

If the MySQL connection code is inconsistent with the actual one, garbled characters may be displayed during parsing on DDM.

In this case, configure **default-character-set=utf8** to specify the encoding system.

Example:

```
mysql -h 127.0.0.1 -P 5066 -D database --default-character-set=utf8 -u ddmuser -p password
```

# 6 Resource Freezing, Release, Deletion, and Unsubscription

---

## Why Are My DDM Resources Released?

If your subscriptions have expired but not been renewed, or if you are in arrears, your resources enter a grace period. If you do not renew the subscriptions or top up your account after the grace period expires, your resources will enter a retention period and become unavailable. If you still do not renew them or top up your account after the retention period ends, your cloud resources will be released and your data stored will be deleted upon the expiration of the retention period. For details, see [Service Suspension and Resource Release](#).

## Why Are My DDM Resources Frozen?

Your resources may be frozen for a variety of reasons. The most common reason is that you are in arrears.

## Can I Back Up Its Data After My Instance Is Frozen?

No. If your instance is frozen because your account is in arrears, go to top up your account to unfreeze your instance and then back up instance data.

## How Do I Unfreeze My DDM Instances?

If your DDM instances are frozen because your account is in arrears, you can unfreeze them by renewing them or topping up your account. The frozen instances can be renewed, released or deleted. Yearly/Monthly DDM instances that have expired cannot be unsubscribed from, while those that have not expired can be unsubscribed from.

## What Impacts Does Resource Freezing, Unfreezing or Release Have on My DDM Instances?

- After a DDM instance is frozen:
  - It cannot be accessed, and your services will be interrupted.
  - No changes can be performed on it if it is a yearly/monthly instance.

- It can be unsubscribed from or deleted manually.
- If data nodes of the instance are frozen, DDM will become unavailable because it cannot communicate with them normally.
- After a frozen DDM instance is unfrozen, you can connect to it again.
- After a DDM instance is released, its relevant resources are also released.

## How Do I Renew My Resources?

When your yearly/monthly DDM instance has expired, you can renew it on the [Renewals](#) page by referring to [Renewal Management](#).

## Can My Resources Be Recovered After They Are Released or Unsubscribed From?

Deleted instances cannot be recovered.

When you unsubscribe from an instance, confirm the instance information carefully. If you have unsubscribed from an instance by mistake, purchase a new one.

## How Do I Delete a DDM Instance?

- For pay-per-use instances, see [Deleting a DDM Instance](#).