Alibaba Cloud ApsaraDB for RDS

RDS for MySQL User Guide

Issue: 20191021

MORE THAN JUST CLOUD | Alibaba Cloud

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

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

Style	Description	Example
{} or {a|b}	**This format is used for a required value, where only one item can be selected.**	`switch {active|stand}`

Contents

Legal disclaimer.....I
Document conventions.....I
1 Preface..... 1
2 Limits of RDS for MySQL.....3
3 Quick start.....5
3.1 General process to use RDS for MySQL.....5
3.2 Create an RDS for MySQL instance.....6
3.3 Configure a whitelist for an RDS for MySQL instance..... 11
3.4 Create accounts and databases for an RDS for MySQL instance.....19
3.5 Connect to an RDS for MySQL instance.....27
3.6 What do I do if I cannot connect an ECS instance to an ApsaraDB for RDS instance?..... 33
3.7 Read-only instances..... 38
3.7.1 Read-only instances overview..... 38
3.7.2 Create an RDS for MySQL read-only instance.....40
3.8 Instance scale-out.....45
4 Data migration and synchronization.....46
4.1 Overview of RDS for MySQL data migration and synchronization..... 46
4.2 Migrate data to an RDS for MySQL instance by using mysqldump..... 47
4.3 Migrate data from a third-party cloud database to ApsaraDB for RDS..... 51
4.3.1 Migrate a MySQL database from Google Cloud to Alibaba Cloud... 51
4.3.2 Migrate MySQL data from AWS RDS to Alibaba Cloud RDS.....58
4.4 Migrate data from a user-created database to ApsaraDB for RDS..... 63
4.5 Migrate data from RDS for MySQL to an on-premises MySQL database.... 63
4.6 Upgrade RDS for MySQL to POLARDB for MySQL with one click..... 67
4.7 Clone data from RDS for MySQL to POLARDB for MySQL with one click... 74
5 Billing management..... 78
5.1 Switch from pay-as-you-go billing to subscription billing..... 78
5.2 Manually renew an RDS for MySQL instance..... 79
5.3 Automatically renew an RDS for MySQL instance..... 82
6 Instance management..... 86
6.1 Restart an RDS instance.....86
6.2 Change the maintenance window of an RDS instance..... 87
6.3 Migrate an RDS for MySQL instance across zones in the same region..... 89
6.4 Manually or automatically switch over services between the RDS for MySQL master and slave instances.....92
6.5 Change the network type of an RDS for MySQL instance..... 96
6.6 Change the data replication mode of an RDS for MySQL instance.....102
6.7 Release an RDS for MySQL instance.....105

6.8 Upgrade the kernel minor version of an RDS for MySQL instance.....107
6.9 Change the configuration of an RDS for MySQL instance..... 109
6.10 Reconfigure parameters for an RDS for MySQL instance.....113
6.11 Operate parameter templates..... 119
6.12 Instance recycle bin..... 126
7 Read-only instances and read/write splitting..... 129
7.1 Read-only instances overview..... 129
7.2 Create an RDS for MySQL read-only instance..... 131
7.3 Set a replication delay for an RDS for MySQL read-only instance.....135
7.4 Introduction to read/write splitting.....137
7.5 Enable read/write splitting for an RDS for MySQL instance..... 140
7.6 Modify the latency threshold and read weights.....146
7.7 Change the network type of the read/write splitting address for an RDS for MySQL instance.....149
7.8 Disable read/write splitting for an RDS for MySQL instance..... 150
7.9 Monitor the read/write splitting performance of an RDS for MySQL instance.....151
7.10 Verify the read/write splitting performance of an RDS for MySQL instance.....152
7.11 Verify the effect of read/write splitting..... 157
7.11.1 Use SQL Explorer to verify the effect of read/Write splitting.... 157
7.11.2 Use internal SQL statements to verify the effect of read/write splitting..... 157
7.12 Verify the allocated read weights.....159
7.13 Rules of weight allocation by the system.....161
7.14 FAQ for read/write splitting.....163
8 Account management.....166
8.1 Create an account for an RDS for MySQL instance.....166
8.2 Reset the password of an account for an RDS for MySQL instance..... 171
8.3 Change the permissions of an account for an RDS for MySQL instance... 172
8.4 Reset the permissions of a premier account.....174
8.5 Authorize a service account for an RDS for MySQL instance..... 175
8.6 Delete an account for an RDS for MySQL instance.....177
9 Database management..... 179
9.1 Create a database for an RDS for MySQL instance..... 179
9.2 Delete a database for an RDS for MySQL instance..... 180
10 Network management.....183
10.1 Switch to a new VPC and VSwitch for an RDS for MySQL instance.....183
10.2 Configure a hybrid access solution to smoothly migrate an RDS instance from the classic network to a VPC..... 185
11 Database connection..... 193
11.1 Configure endpoints for an RDS for MySQL instance..... 193
11.2 Log on to an RDS for MySQL instance through DMS.....197

11.3 Apply for a public endpoint for an RDS for MySQL instance.....198
11.4 View the internal and public endpoints and ports of an RDS for MySQL instance.....202
12 Dedicated database proxy.....203
13 Monitoring and alerts.....212
13.1 View resource and engine monitoring data.....212
13.2 Set the monitoring frequency.....215
13.3 Set an alert rule.....217
14 Data security.....220
14.1 Switch to the enhanced whitelist mode for an RDS for MySQL instance..220
14.2 Configure a whitelist for an RDS for MySQL instance.....222
14.3 Configure SSL encryption for an RDS for MySQL instance.....230
14.4 Set TDE for an RDS for MySQL instance.....236
15 SQL audit and historical events.....239
15.1 SQL audit.....239
15.2 SQL Explorer.....239
16 Database backup.....247
16.1 Back up the data of an RDS for MySQL instance.....247
16.2 View the quota of free backup space for an RDS for MySQL instance....253
16.3 Download the data backup files and log backup files of an RDS for MySQL instance.....255
16.4 Configure local binary log policies for an RDS for MySQL instance.....258
16.5 Back up data across regions.....262
17 Database restoration.....269
17.1 Restore the data of an RDS for MySQL instance.....269
17.2 Restore individual databases or tables for an RDS for MySQL instance..278
17.3 Restore the data of an RDS for MySQL instance across regions.....285
18 Disable the database proxy mode.....289
19 Manage logs.....292
20 Tag management.....294
20.1 Create tags.....294
20.2 Delete tags.....296
20.3 Filter RDS instances by tag.....297
21 Appendixes.....299
21.1 Commonly used SQL statements for MySQL.....299
21.2 Grant backup file download permissions to a RAM user with only read-only permissions.....300
21.3 Cached data persistence.....302
21.4 Multi-structure data storage.....303

1 Preface

This topic provides an overview of RDS for MySQL, including a disclaimer, terms, and concepts.

Overview

Alibaba Cloud ApsaraDB for RDS (short for Relational Database Service) is a stable, reliable, and scalable online database service. Based on Alibaba Cloud distributed file system and high-performance SSD storage, ApsaraDB for RDS supports the MySQL, SQL Server, PostgreSQL, and PPAS (compatible with Oracle) database engines and provides a portfolio of solutions to disaster tolerance, backup, recovery, monitoring, and migration to facilitate database operation and maintenance. For information about the benefits of ApsaraDB for RDS, see *#unique_4*.

This document describes how to configure and manage RDS through the *RDS console* , helping you better understand the features and functions of ApsaraDB for RDS. Additionally, you can configure and manage RDS through API and SDK.

If you need technical support, you can call 95187. Alternatively, you can open the *RDS console* and in the upper-right corner choose More > Support > Open a new ticket. If your business is complex, you can purchase a *support plan* to obtain your exclusive support service from IM enterprise groups, technical service managers (TAM), and service managers.

For more information about ApsaraDB for RDS, visit *ApsaraDB RDS for MySQL*.

Disclaimer

Some product features or services described in this document may be unavailable for certain regions. See the relevant commercial contracts for specific Terms and Conditions. This document serves as a user guide only. No content in this document can constitute any express or implied warranty.

Terms

- **Instance: A database service process that takes up physical memory independently. You can specify the memory size, disk space, and database type of an instance, but only the memory specification determines the performance of the**

instance. After an instance is created, you can delete it or change its configurat ion as needed.

- Database: A logical unit created in an instance. Multiple databases that each have a unique name can be created in one instance.
- Region and zone: A region is a physical data center. A zone is a physical area that has independent power supply and network in a region. For more information, visit *Alibaba Cloud's Global Infrastructure*.

Concepts

Item	Description
On-premises database	A database that is deployed in your on-premises equipment room or on a cloud other than ApsaraDB for RDS.
RDS for XX (XX is MySQL, SQL Server, PostgreSQL, or PPAS.)	A type of RDS instance. For example , RDS for MySQL refers to the type of RDS instance that runs in the MySQL database engine.

2 Limits of RDS for MySQL

This topic describes the limits of RDS for MySQL. To guarantee stability and security, you must understand the limits.

The following table describes the limits of RDS for MySQL.

Items	Restrictions
Parameter modification	The *RDS console* or APIs must be used to modify database parameters. But some parameters cannot be modified. For more information, see *#unique_6*.
Root permission	The root or sa permission is not provided.
Backup	· Command lines or graphical interfaces can be used for logical backup. · For physical backup, the *RDS console* or APIs must be used.
Restoration	· Command lines or graphical interfaces can be used for logical restoration. · For physical restoration, the *RDS console* or APIs must be used.
Migration	· Command lines or graphical interfaces can be used for logical import. · You can use the MySQL command line tool or Data Transmissi on Service (DTS) to migrate data.
MySQL storage engine	· Currently only InnoDB and TokuDB are supported. The MyISAM engine has defects and may cause data loss. If you create MyISAM engine tables, they are automatically converted to InnoDB engine tables. For more information, see *Why does RDS for MySQL not support the MyISAM engine?* · The InnoDB storage engine is recommended for performance and security requirements. · The Memory engine is not supported. If you create Memory engine tables, they are automatically converted to InnoDB engine tables.

Items	Restrictions
Replication	MySQL provides a dual-node cluster based on the master/slave replication architecture, so you manual deployment is not required. The slave instance in the architecture is invisible to you, and your application cannot access to the slave instance directly.
Restarting RDS instances	Instances must be restarted through the *RDS console* or APIs.
User, password, and database management	By default, *RDS console* is used to manage users, passwords, and databases, including operations such as instance creation, instance deletion, permission modification, and password modification. MySQL also allows you to create a master account for finer-grained management.
Common account	· Does not support customized authorization. · The account management and database management interfaces are provided on the RDS console. · Instances that support common accounts also support master accounts.
Master account	· Support customized authorization. · SQL statements can be used for management.
Network settings	If a MySQL 5.5/5.6 instance is in a classic network and its *access mode* is safe connection mode, do not enable net.ipv4.tcp_timestamps in SNAT mode.

3 Quick start

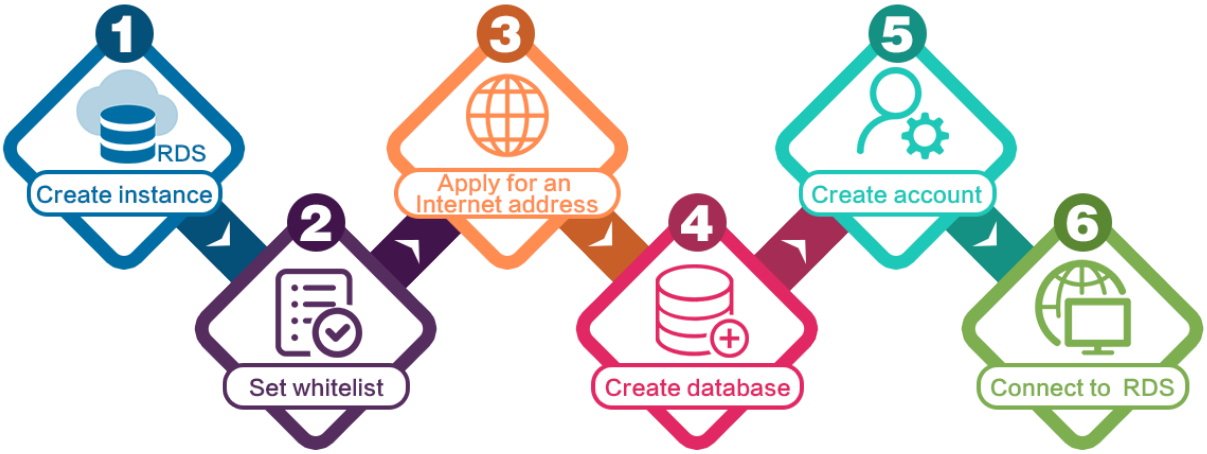
3.1 General process to use RDS for MySQL

This topic describes the general process from purchasing an RDS for MySQL instance to using it, including creating, setting, and connecting an instance.

Quick Start flowchart

If this is the first time that you use RDS for MySQL, read *Limits of RDS for MySQL* before you purchase an RDS for MySQL instance.

The following flowchart shows the operations you must complete from purchasing an RDS for MySQL instance to using it.



1. *Create an RDS for MySQL instance*
2. *Configure a whitelist for an RDS for MySQL instance*
3. *Apply for a public endpoint for an RDS for MySQL instance*
4. *Create accounts and databases for an RDS for MySQL instance*
5. *Connect to an RDS for MySQL instance*

3.2 Create an RDS for MySQL instance

This topic describes how to create an RDS for MySQL instance through the RDS console.

For information about how to create an RDS instance in other DB engines, see the following resources:

- *#unique_15*
- *#unique_16*
- *#unique_17*
- *#unique_18*

If you want to create an RDS for MySQL instance by calling an API action, see *CreateDBInstance*.

For information about the pricing of RDS for MySQL instances, see *#unique_20*.

Prerequisites

You have registered an Alibaba Cloud account.

For more information, see *Sign up with Alibaba Cloud*.

By

Precautions

- **Subscription instances cannot be converted to pay-as-you-go instances.**
- **Pay-as-you-go instances can be converted to subscription instances. For more information, see *Switch from pay-as-you-go billing to subscription billing*.**
- **By default, each Alibaba Cloud account can create up to 30 pay-as-you-go RDS instances. You can *open a ticket* to apply for an increase to the limit.**

Procedure

1. **Log on to the *RDS console*.**
2. **On the Instances page, click Create Instance.**

3. Select a billing method:
   - Pay-As-You-Go: indicates post payment (billed by hour). For short-term requirements, create pay-as-you-go instances because they can be released at any time to save costs.
   - Subscription: indicates prepayment. You must pay when creating an instance. For long-term requirements, create subscription instances because they are more cost-effective. Furthermore, the longer the subscription, the higher the discount.
4. Set the following parameters.

Parameter	Description
Region	Select the region in which the RDS instance to be purchased will be located. The region cannot be changed after the instance is created. We recommend that you: · Select the same region as the corresponding ECS instance to avoid incurring charges for Internet traffic usage and guarantee fast access. · Check whether the selected region supports your required MySQL version and whether multi-zone support is available.
Database Engine	Select a DB engine. In this example, select MySQL. Note: The available DB engines vary depending on the region you select.
Version	Select a version of MySQL. You can select MySQL 5.5, 5.6, 5.7, or 8.0. Note: The available versions vary depending on the region you select.

Parameter	Description
Edition	Select an RDS edition. Valid values: · Basic: The DB system has only one instance. In this edition, computation is separated from storage, which is cost-effective. However, we recommend that you do not use this edition in production environments. · High-availability: The DB system has two instances: one master instance and one slave instance. The two instances work in a classic high-availability architecture. · Enterprise Edition: The DB system has three instances: one master instance and two slave instances. The three instances are located in three different zones in the same region to guarantee service availability. This edition is available to the China (Hangzhou), China (Shanghai), China (Shenzhen), and China (Beijing) regions. Note: The available editions vary depending on the DB engine version you select. For information about the RDS editions, see #unique_22.
Storage Type	Select a storage type. Valid values: · Local SSD: An SSD that is located on the same node as the DB engine. Storing data to local SSDs reduces I/O latency. · Standard SSD: An elastic block storage device that is designed based on a distributed storage architecture. Storing data to cloud SSDs makes separation between computation and storage possible. · Enhanced SSD: An SSD that is designed based on the new-generation distributed block storage architecture and the 25 GB and RDMA technologies to reduce single-link latency. Each enhanced SSD can process up to 1,000,000 random read and write requests. For more information, see #unique_23.

Parameter	Description
Zone	Select a zone. A zone is a physical area within a region. Different zones in the same region are basically the same. You can deploy the master and slave instances in the same zone or in different zones. Multi-zone deployment is more secure because it provides zone-level disaster tolerance.
Network Type	Select a network type. Valid values: · Classic Network: indicates a traditional network. · VPC (recommended): short for Virtual Private Cloud. A VPC is an isolated network environment and therefore provides higher security and performance than a classic network. Note: Make sure the network type of the RDS instance is the same as that of your ECS instance so that the ECS instance can access the RDS instance through the intranet.
Type	Select an RDS instance type. The RDS instance type specifies the specifications of the RDS instance. Each type supports a specific number of CPU cores, memory size, maximum number of connections, and maximum IOPS. For more information, see *#unique_24*. Valid values: · General-purpose instance: provides dedicated memory and I/O resources, but shares the CPU and storage resources with the other general-purpose instances on the same server. · Dedicated instance: provides dedicated CPU, memory, storage, and I/O resources. · Dedicated host: provides all the CPU, memory, storage, and I/O resources on the server where it is located. For example, 8 Cores 32 GB (Basic) indicates a general-purpose instance, and 8 Cores 32 GB (Dedicated) indicates a dedicated instance.

Parameter	Description
Capacity	The capacity is used for storing data, system files, binlog files, and transaction files.

5. Optional. Set the duration of the billing method for a subscription instance and specify the number of instances to be created. Then, click Buy Now.

> **Note:**
> For a subscription instance, you can:
>
> - Select Auto Renew in the Duration section. Then the system can automatically deduct fees to extend the validity period of the instance. For example, if you purchase a three-month subscription instance with Auto Renew selected, the system automatically deducts frees of three months when the instance is about to expire.
> - Click Add to Cart and then click the cart to place the order.

6. On the Order Confirmation page, read and confirm you agree to Terms of Service, Service Level Agreement, and Terms of Use by selecting the checkbox, confirm the order details, and click Pay Now.

What to do next

Log on to the *RDS console*, select the target region, and view the instance details.

After the RDS instance is created, you must *configure whitelists* and *create accounts* for it. If you want to connect to the RDS instance through the Internet, you must also *apply for a public endpoint* for it. After all is done, you can *connect to the RDS instance*.

FAQ

How do I authorize a RAM user to manage RDS instances?

See *#unique_25*.

APIs

API	Description
#unique_26	Used to create an RDS instance.

3.3 Configure a whitelist for an RDS for MySQL instance

This topic describes how to configure a whitelist for an RDS for MySQL instance. After you create an RDS instance, you must configure a whitelist for it to allow external devices to access the instance.

RDS for PostgreSQL provides two types of whitelists::

- **IP address whitelist: Add IP addresses to a whitelist so that these IP addresses can access the RDS instance.**

  Note:
  The default whitelist is an IP address whitelist that contains only the default IP address 127.0.0.1. This default IP address means that no devices can access the RDS instance.

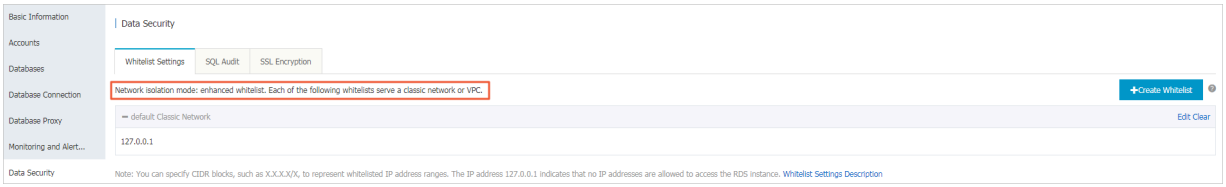
- **VPC security group whitelist: Add a VPC security group to a whitelist so that all ECS instances in the VPC security group can access the RDS instance.**

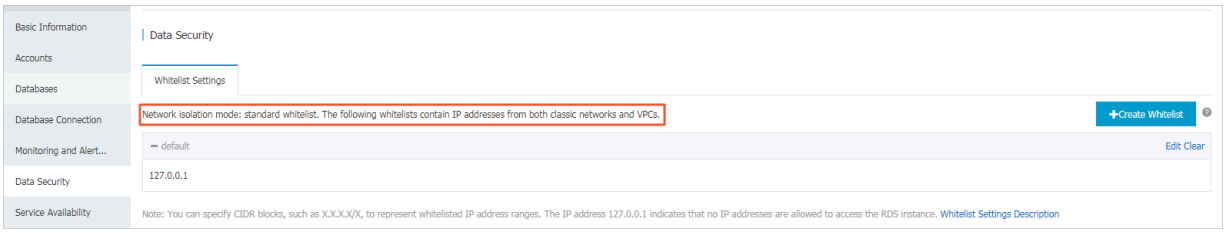
A whitelist can be used to improve the security of your RDS instance. We recommend that you periodically check and adjust your whitelists to maintain RDS security. Configuring a whitelist does not affect the normal running of the RDS instance.

Configure an IP address whitelist

Precautions

- **The default IP whitelist can only be edited or cleared, but cannot be deleted.**
- **If you log on to DMS but the DMS server IP address has not been added to the whitelist, DMS prompts you to add the IP address and automatically generates a whitelist containing the IP address.**
- **You must confirm which network isolation mode your RDS instance is in before configuring the whitelist. See the corresponding operations based on the network isolation mode.**





Note:

The private networks to which RDS instances belong are divided into two types: Classic Network and VPC.

- **Classic network: Alibaba Cloud automatically allocates IP addresses. Users only need to perform simple configurations. This network type is suitable for new users.**
- **VPC: You can customize the network topology and IP addresses. It supports leased line connection, and is suitable for advanced users.**

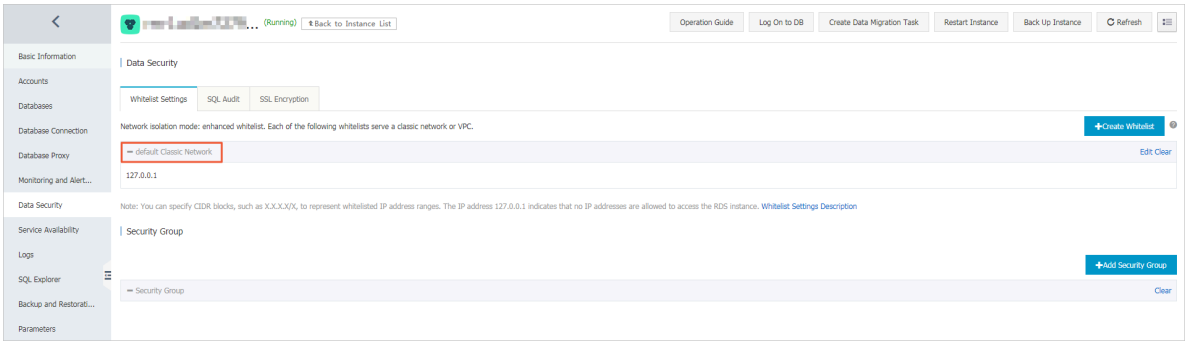
If the network isolation mode is enhanced whitelist mode

1. **Log on to the *RDS console*.**
2. **In the upper-left corner of the page, select the region where the target instance is located.**

3. **Find the target instance and click its ID.**
4. **In the left-side navigation pane, click Data Security.**

5. On the Whitelist Settings tab page, follow these instructions based on your scenario:
   - To access the RDS instance from an ECS instance located within a VPC, click Edit for `default VPC` whitelist.
   - To access the RDS instance from an ECS instance located within the classic network, click Edit for the `default Classic Network` whitelist.
   - To access the RDS instance from a server or computer located in a public network, click Edit for the `default Classic Network` whitelist.

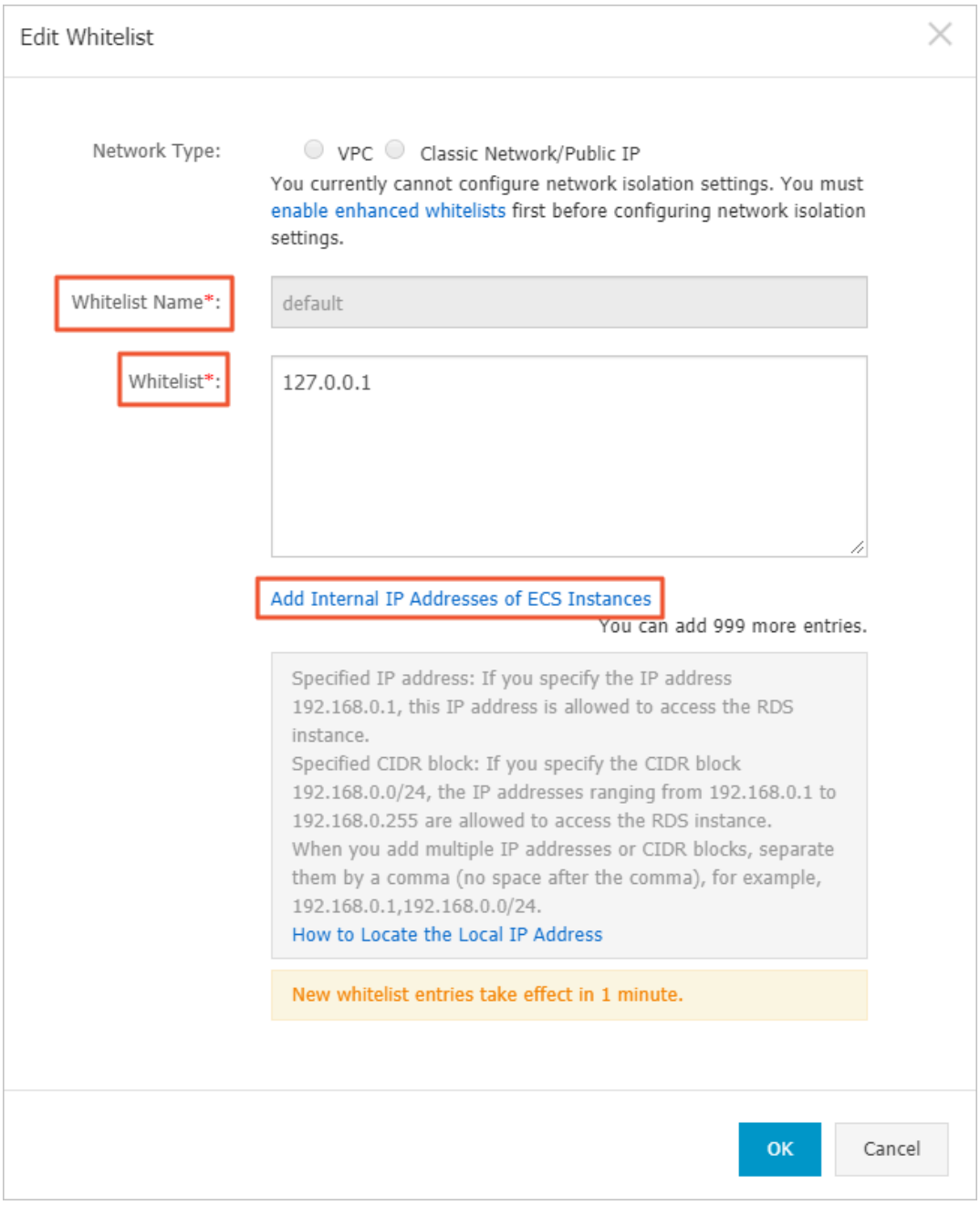
   Note:
   - To allow ECS to access RDS through the intranet (VPC or classic network), make sure that the two instances are in the same region and have the same *network type*. Otherwise, the connection fails.
   - You can also click Create Whitelist to create a new whitelist. In the displayed Create Whitelist dialog box, select a network type: VPC or Classic Network/ Public IP.



6. In the displayed Edit Whitelist dialog box, add the IP addresses that need to access the instance, and then click OK.
   - If you add an IP address range, such as 10.10.10.0/24, any IP address in the 10.10.10.*X* format can access the RDS instance.
   - To add multiple IP addresses or IP address ranges, separate them with a comma (without spaces), for example, `192.168.0.1,172.16.213.9`.
   - You can click Add Internal IP Addresses of ECS Instances to display the IP addresses of all the ECS instances under your Alibaba Cloud accoun. and add to the whitelist.

Note:

After you add any IP address to the default whitelist, the default IP address 127.0.0.1 is automatically removed.

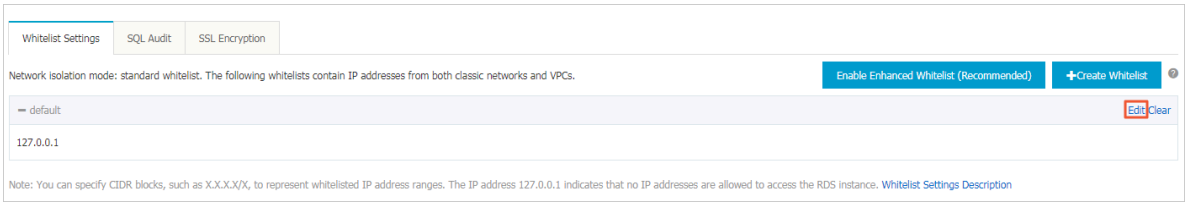


If the network isolation mode is standard whitelist mode

1. **Log on to the *RDS console*.**
2. **In the upper-left corner of the page, select the region where the target instance is located.**
3. **Find the target instance and click its ID.**
4. **In the left-side navigation pane, click Data Security.**

5. On the Whitelist Settings tab page, click Edit for the default whitelist.

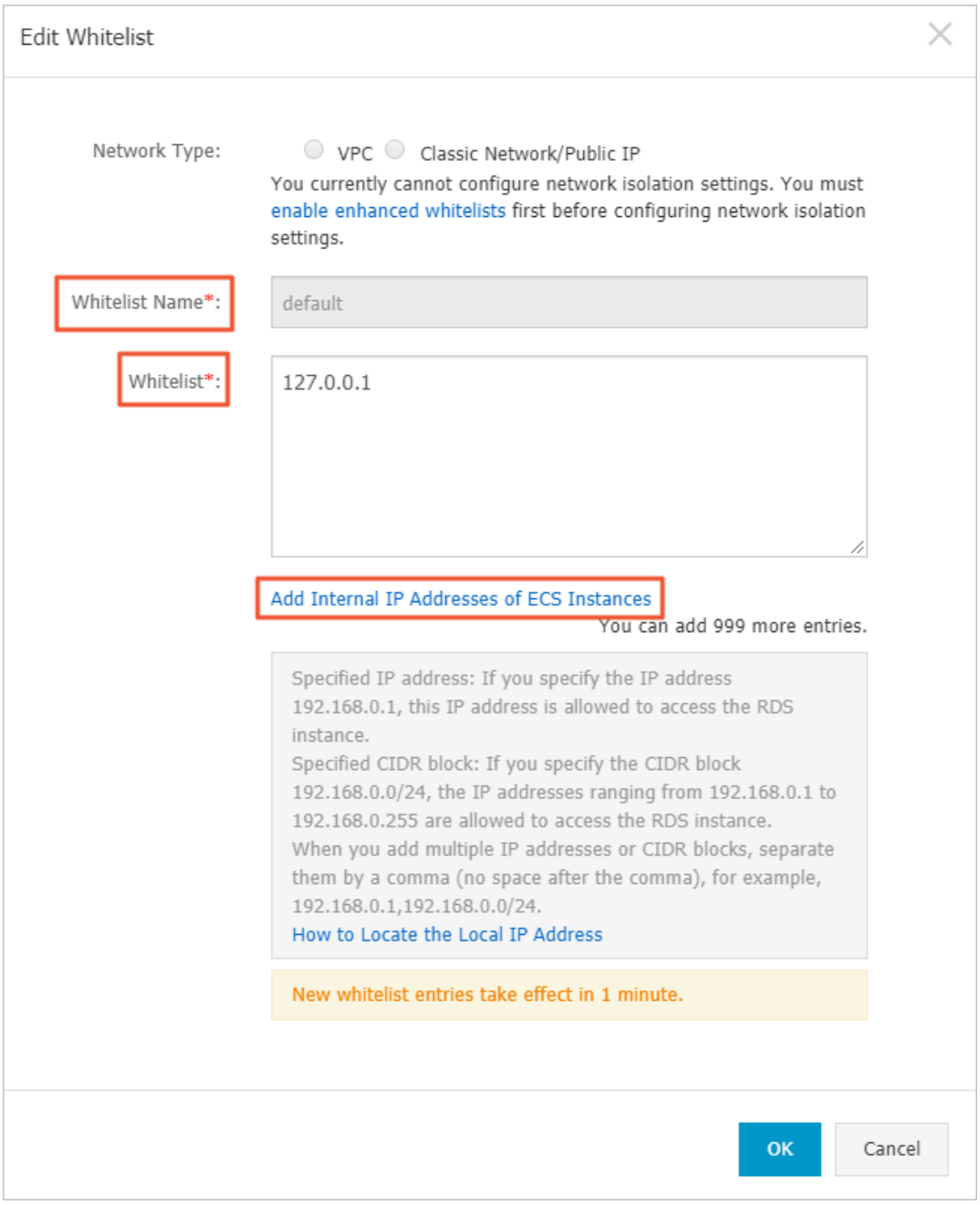
> **Note:**
> You can also click Create Whitelist to create another whitelist.



6. In the displayed Edit Whitelist dialog box, add the IP addresses that need to access the instance, and then click OK.
   - If you add an IP address range, such as 10.10.10.0/24, any IP address in the 10.10.10.*X* format can access the RDS instance.
   - To add multiple IP addresses or IP address ranges, separate them with a comma (without spaces), for example, `192.168.0.1,172.16.213.9`.
   - You can click Add Internal IP Addresses of ECS Instances to display the IP addresses of all the ECS instances under your Alibaba Cloud accoun. and add to the whitelist.

> **Note:**

After you add any IP address to the default whitelist, the default address 127.0.0.1 is automatically removed.



Common incorrect settings

- **If the IP address whitelist contains only the default address 127.0.0.1, no device is allowed to access the RDS instance. Therefore, you must add IP addresses of devices to the whitelist to allow access to the instance.**
- **The IP address in the whitelist is 0.0.0.0, but the correct format is 0.0.0.0/0.**

Note:

0.0.0.0/0 indicates that all devices are allowed to access the RDS instance. Exercise caution when using this IP address.

- **If the *enhanced whitelist* mode is enabled, check the following:**
  - **To connect to the RDS instance through the VPC address of the RDS instance, make sure that the private IP address of the ECS instance is added to the default VPC whitelist.**
  - **To connect to the RDS instance through the classic network address of the RDS instance, make sure that the private IP address of the ECS instance is added to the default Classic Network whitelist.**
  - **To connect to the RDS instance through *ClassicLink*, make sure that the private IP address of the ECS instance is added to the default VPC whitelist.**
  - **To connect to the RDS instance through the public address of the RDS instance, make sure that the public IP address of the device is added to the default Classic Network whitelist.**
- **The public IP address that you add to the whitelist may not be the real outbound IP address. The reasons are as follows:**
  - **The public IP address is not fixed and may dynamically change.**
  - **The tools or websites used to query the public IP addresses may provide wrong IP addresses.**

  For more information, see *#unique_29*.

Configure an ECS security group

An ECS security group is a virtual firewall that is used to control the inbound and outbound traffic of ECS instances in the security group. After an ECS security group is added to the RDS whitelist, the ECS instances in the security group can access the RDS instance.

For more information, see *Create a security group*.

Precautions

- **RDS versions that support an ECS security group are MySQL 5.6 and 5.7.**
- **You can configure both the IP address whitelists and an ECS security group. The IP addresses in the whitelists and the ECS instances in the security group can all access the RDS instance.**
- **You can only add one ECS security group to an RDS instance.**

- **Updates to the ECS security group are automatically synchronized to the whitelist in real time.**

Procedure

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target instance is located.**
3. **Find the target instance and click its ID.**
4. **In the left-side navigation pane, click Data Security.**
5. **On the Whitelist Settings tab page, click Add Security Group.**

   Note:
   Security groups with a VPC tag are within VPCs.
6. **Select a security group and click OK.**

APIs

API	Description
#unique_30	**Used to view the IP address whitelist of an RDS instance.**
#unique_31	**Used to modify the IP address whitelist of an RDS instance.**

3.4 Create accounts and databases for an RDS for MySQL instance

This topic describes how to create accounts and databases for an RDS for MySQL instance.

For information about how to create accounts and databases in other DB engines, see the following resources:

- *#unique_32*
- *#unique_33*
- *#unique_34*
- *#unique_35*
- *#unique_36*
- *#unique_37*

Account types

RDS for MySQL supports two types of database accounts: premier accounts and standard accounts. You can manage all your accounts and databases through the RDS console. For information about specific permissions, see *Supported database permissions according to account type* **.**

Account type	Description
Premier account	· Can only be created and managed through the RDS console or API. · Each RDS for MySQL instance can have only one premier account, which can be used to manage all databases and standard accounts. · Has more permissions than standard accounts and can manage permissions at a more fine-grained level. For example, it can assign table-level query permissions to other accounts. · Can disconnect the connections established by any other accounts.
Standard account	· Can be created and managed through the RDS console, API, or SQL statements. · Each RDS for MySQL instance can have up to 200 standard accounts. · Must be manually authorized with database permissions. · Cannot create or manage other accounts, or terminate the connections established by other accounts.

Account Type	Number of databases	Number of tables	Number of users
Premier account	Unlimited	< 200,000	Varies depending on the instance kernel parameters.
Standard account	500	< 200,000	Varies depending on the instance kernel parameters.

Differences between the premier account permissions and SUPER permissions

Premier account permissions

- **Can manage all databases and standard accounts.** *Supported database permissions according to account type* **lists the permissions of the premier account.**
- **Can terminate the connections established by other accounts.**
- **Running the** `show processlist` **command shows only the processes of the current account, excluding processes on the control level.**

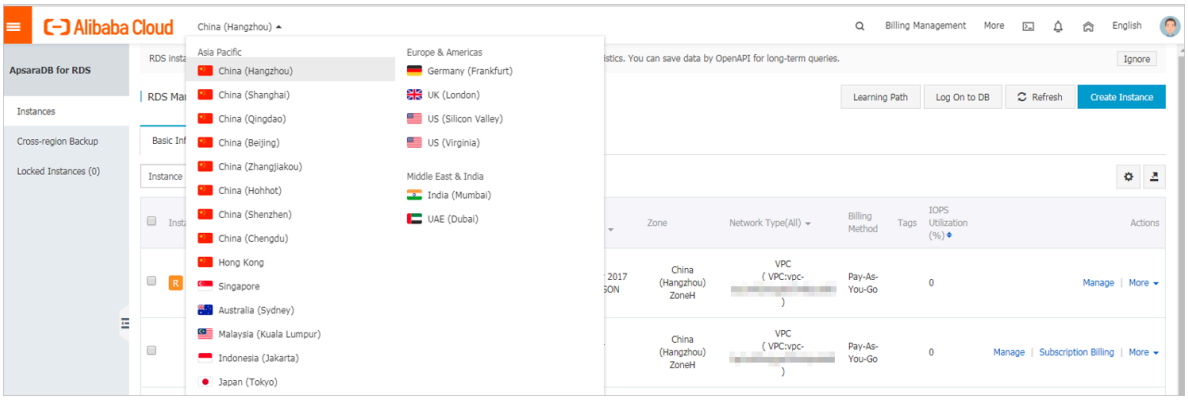
SUPER permissions

To prevent potential incorrect operations, RDS for MySQL does not provide the SUPER permissions.

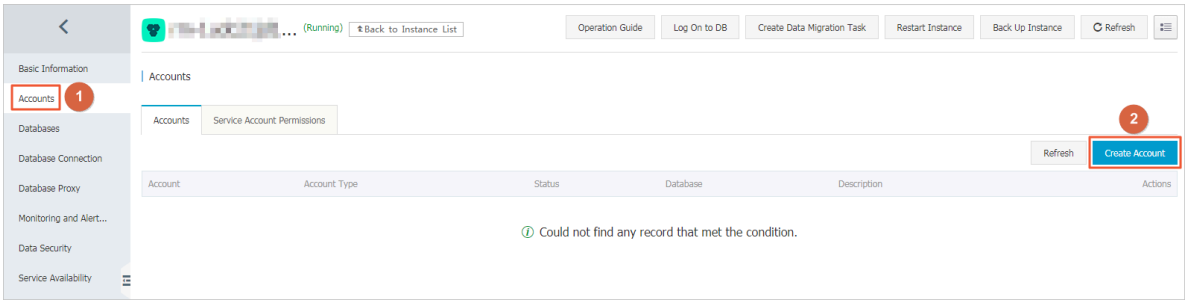
- **Can terminate any connections.**
- **Running the `show processlist` command shows all processes, including processes on the control level.**
- **Can use the `SET` statement to modify any global variables.**
- **Can use the `CHANGE MASTER` and `PURGE MASTER LOGS` commands.**
- **Can perform operations on files stored on the host.**

Create the premier account

1. **Log on to the *RDS console*.**
2. **Select the target region.**



3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Accounts.**
5. **Click Create Account.**



6. Set the following parameters.

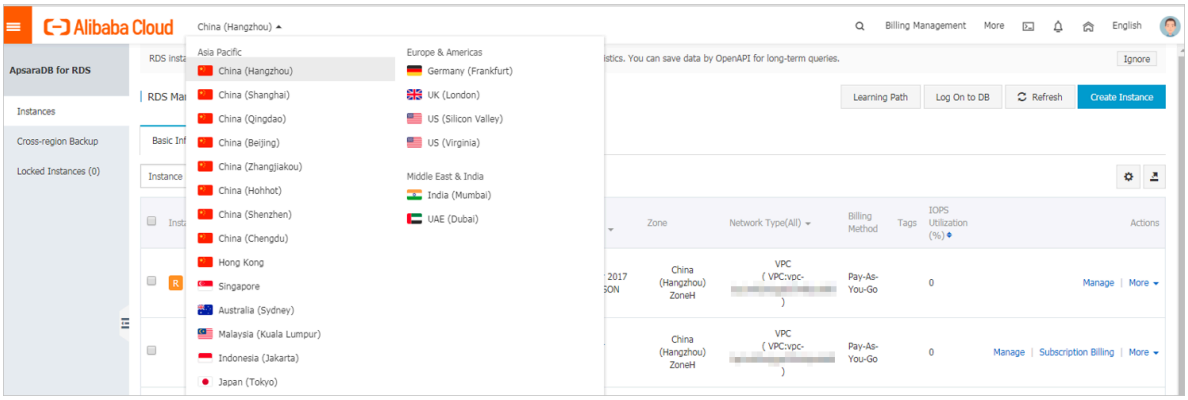
Parameter	Description
Database Account	The account name must be 2 to 16 characters in length and can contain lowercase letters, numbers, and underscores (_). It must start with a letter and end with a letter or number.
Account Type	Select Premier Account.
Password	The account password must be 8 to 32 characters in length and contain at least three of the following types of characters : uppercase letters, lowercase letters, numbers, and special characters. The allowed special characters are as follows: ! @ # $ % ^ & * ( ) _ + - =
Re-enter Password	Enter the password again.
Note	Optional. Enter details about the premier account to better identify it. You can enter up to 256 characters.

7. Click OK.

Reset the permissions of the premier account

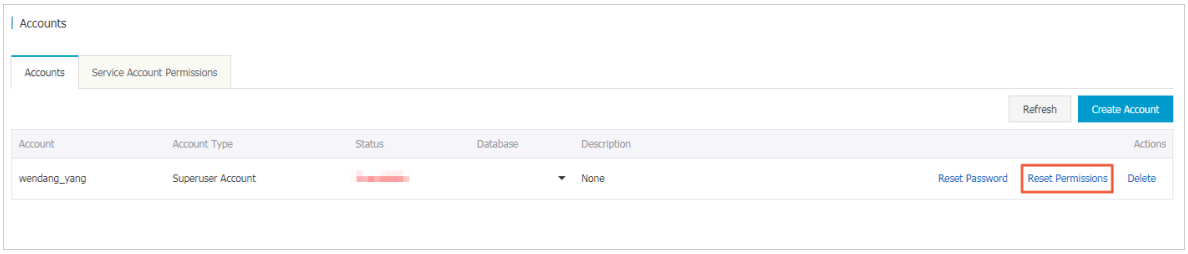
If the premier account is abnormal (for example, the account permissions are unexpectedly revoked), you can reset the permissions. To do so, follow these steps:

1. Log on to the *RDS console*.
2. Select the target region.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Accounts.

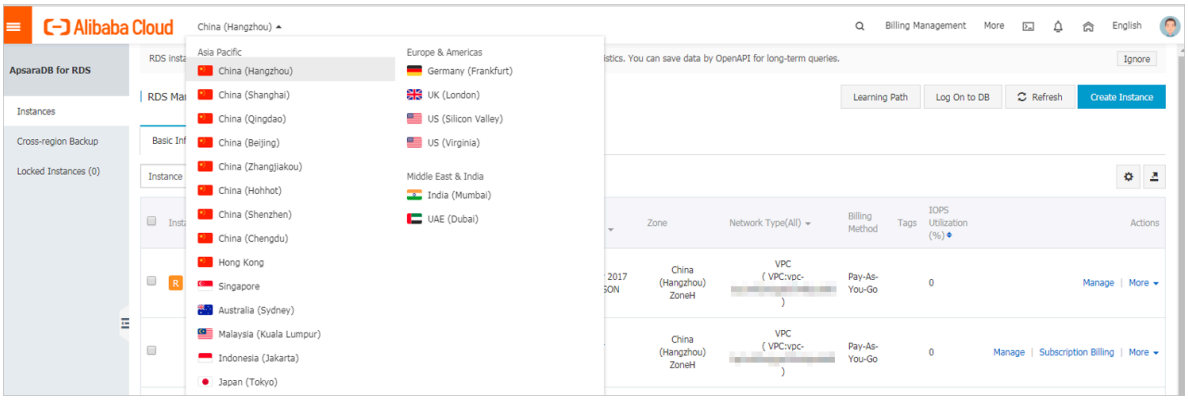
5. Find the premier account and in the Actions column click Reset Permissions.



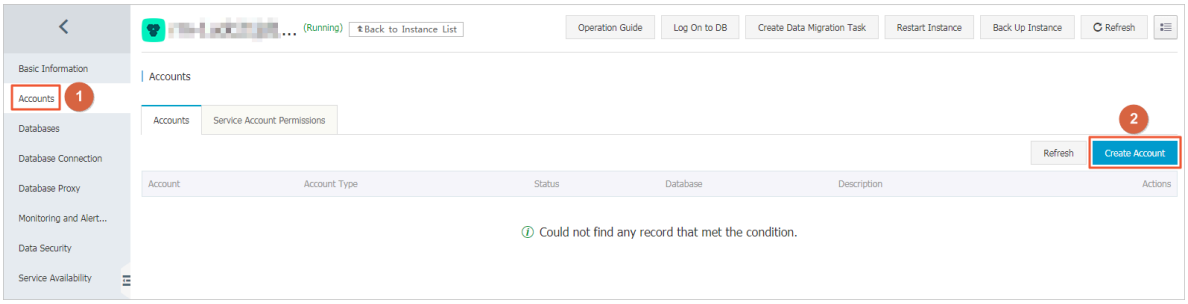
6. Enter the password of the premier account and click OK.

Create a standard account

1. Log on to the *RDS console*.
2. Select the target region.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Accounts.
5. Click Create Account.



6. Set the following parameters.

Parameter	Description
Database Account	The account name must be 2 to 16 characters in length and can contain lowercase letters, numbers, and underscores (_). It must start with a letter and end with a letter or number.
Account Type	Select Standard Account.

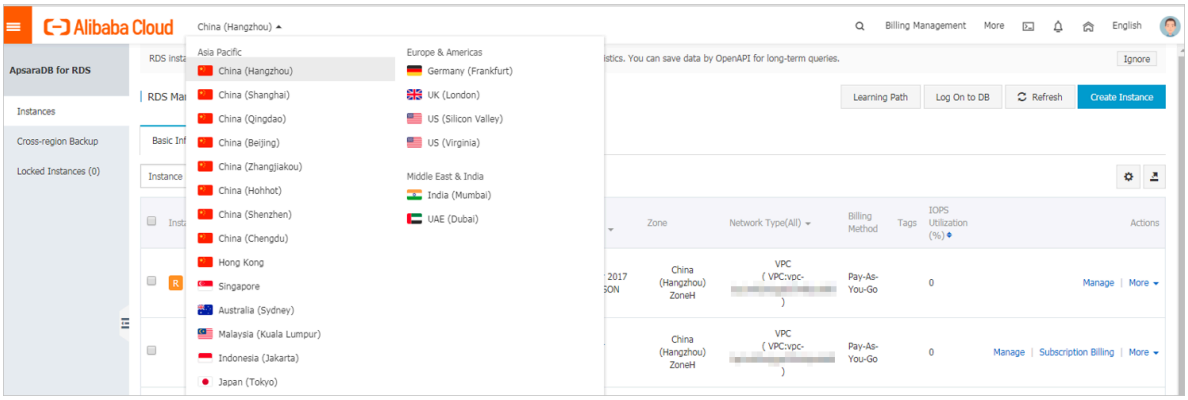
Parameter	Description
Authorized Databases	Grant permissions on one or more databases to the account. This parameter is optional. You can also grant permissions to the account after the account is created. a. Select one or more databases from the left area and click Authorize > to add the selected databases to the right area. b. In the right area, click Read/Write, Read-only, DDL Only, or DML Only. If you want to grant the permissions for multiple databases in batches, select all the databases and in the upper-right corner click the button such as Full Control Read/Write. Note: The button in the upper-right corner changes as you click. For example, after you click Full Control Read/Write, the permission changes to Full Control Read-only.
Password	The account password must be 8 to 32 characters in length can contain at least three of the following types of characters : uppercase letters, lowercase letters, numbers, and special characters. The allowed special characters are as follows: ! @ # $ % ^ & * ( ) _ + - =
Re-enter Password	Enter the password again.
Note	Optional. Enter details about the standard account to better identify it. You can enter up to 256 characters.

7. Click OK.

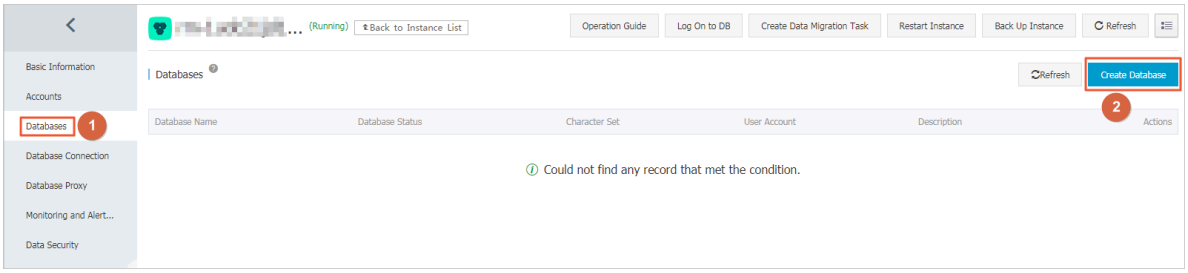
Create a database

1. Log on to the *RDS console*.

2. Select the target region.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Databases.
5. Click Create Database.



6. Set the following parameters.

Parameters	Description
Database Name	The account name must be 2 to 16 characters in length and can contain lowercase letters, numbers, underscores (_), and hyphens (-). It must start with a letter and end with a letter or number.
Supported Character Set	Select utf8, gbk, latin1, or utf8mb4. If you need another character set, select all and then select the character set from the list.
Authorized Account	Select the account that needs to access this database. You can also leave this parameter blank and set the authorized account after the database is created. **Note:** Only standard accounts are displayed, because the premier account has all permissions for all databases.
Account Type	Select Read/Write, Read-only, DDL only, or DML only.

Parameters	Description
Description	Optional. Enter details about the database to better identify it. You can enter up to 256 characters.

7. Click OK.

Supported database permissions according to account type

Account type	Permiss type	Permission				
Premier account	N/A	SELECT	INSERT	UPDATE	DELETE	CREATE
		DROP	RELOAD	PROCESS	REFERENCES	INDEX
		ALTER	CREATE TEMPORARY TABLES	LOCK TABLES	EXECUTE	REPLICATION SLAVE
		REPLICATION CLIENT	CREATE VIEW	SHOW VIEW	CREATE ROUTINE	ALTER ROUTINE
		CREATE USER	EVENT	TRIGGER	N/A	N/A
Standard account	Read-only	SELECT	LOCK TABLES	SHOW VIEW	PROCESS	REPLICATION SLAVE
		REPLICATION CLIENT	N/A	N/A	N/A	N/A
	Read/write	SELECT	INSERT	UPDATE	DELETE	CREATE
		DROP	REFERENCES	INDEX	ALTER	CREATE TEMPORARY TABLES
		LOCK TABLES	EXECUTE	CREATE VIEW	SHOW VIEW	CREATE ROUTINE
		ALTER ROUTINE	EVENT	TRIGGER	PROCESS	REPLICATION SLAVE
		REPLICATION CLIENT	N/A	N/A	N/A	N/A
	DDL only	CREATE	DROP	INDEX	ALTER	CREATE TEMPORARY TABLES
		LOCK TABLES	CREATE VIEW	SHOW VIEW	CREATE ROUTINE	ALTER ROUTINE

Account type	Permiss type	Permission				
		PROCESS	REPLICATION SLAVE	REPLICATION CLIENT	N/A	N/A
	DML only	SELECT	INSERT	UPDATE	DELETE	CREATE TEMPORARY TABLES
		LOCK TABLES	EXECUTE	SHOW VIEW	EVENT	TRIGGER
		PROCESS	REPLICATION SLAVE	REPLICATION CLIENT	N/A	N/A

FAQ

Can I use an account that I create to operate read-only instance?

An account created in your master RDS instance is synchronized to the read-only instances attached to the master instance. However, you cannot manage the account in the read-only instances. The account only has the read permissions for the read-only instances.

APIs

API	Description
#unique_38	Used to create an account for an RDS instance.
#unique_39	Used to create a database for an RDS instance.

3.5 Connect to an RDS for MySQL instance

This topic describes how to connect to an RDS for MySQL instance. After completing the initial configuration, you can connect to your RDS instance from an ECS instance or your computer.

For information about how to create accounts and databases in other DB engines, see the following resources:

- *#unique_15*
- *#unique_40*

- *#unique_41*
- *#unique_42*

Background information

After you *create an instance*, *configure a whitelist*, and *create an account*, you can use DMS, a database client, or CLI to connect to your RDS instance. You can also set the IP address, port, and account information in applications to connect.

Use DMS to connect to an RDS instance

DMS is a graphical data management service provided by Alibaba Cloud. It can be used to manage non-relational databases and relational databases, and supports data and schema management, user authorization, security audit, data trends, data tracking, BI charts, and performance and optimization.

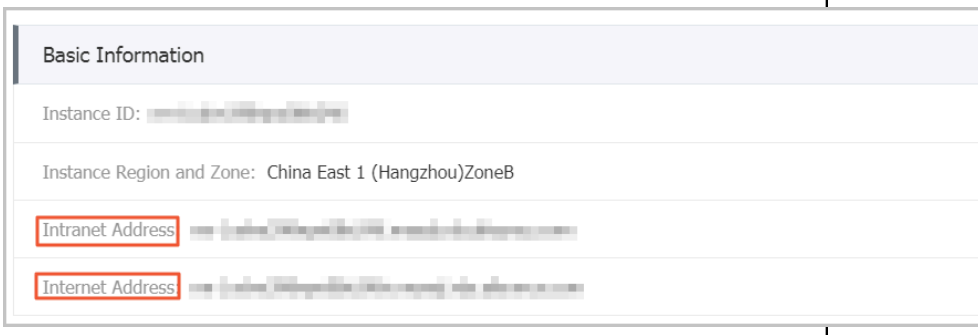
For more information, see *Use DMS to log on to an RDS instance*.

Use a database client to connect to an RDS instance

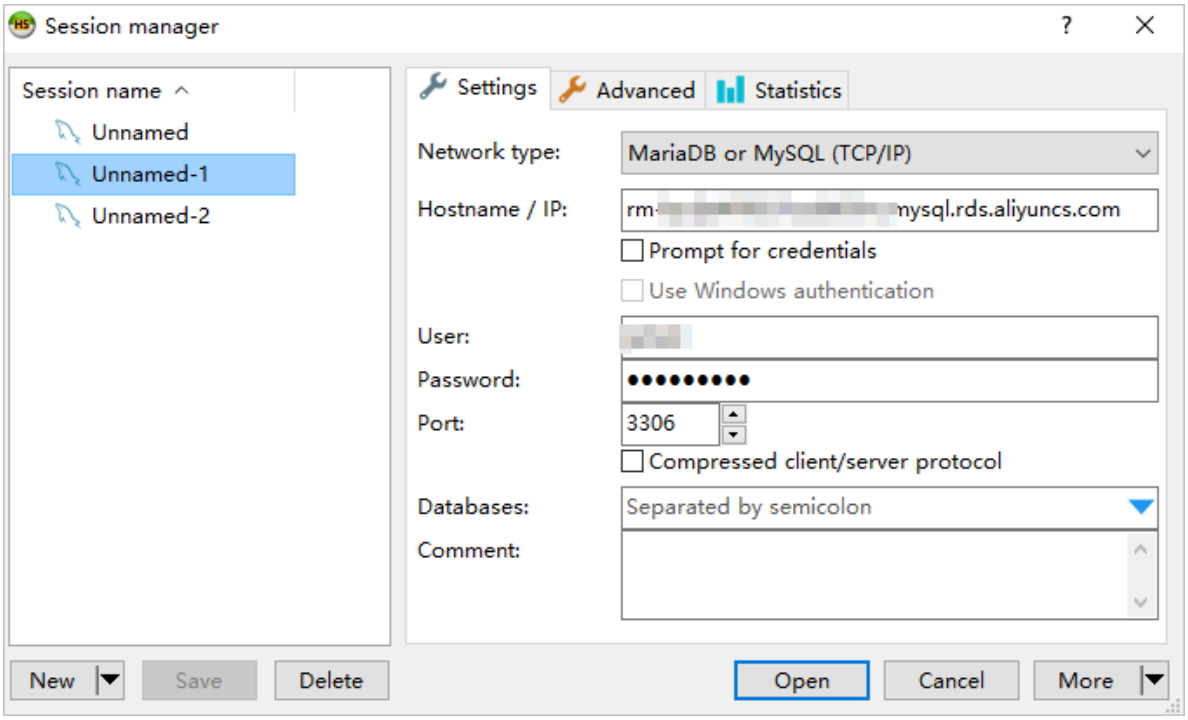
ApsaraDB RDS for MySQL is fully compatible with MySQL. You can connect to an RDS instance from any general-purpose database client in the similar way you connect to a MySQL database. This section describes how to use *HeidiSQL* to connect to an RDS instance.

1. **Start HeidiSQL.**
2. **In the lower-left area of the Session manager dialog box, click New.**
3. **Enter the information of the RDS instance to be connected. The following table describes the parameters.**

Parameter	Description
Network type	The method of connecting to the RDS instance. Select MariaDB or MySQL (TCP/IP).

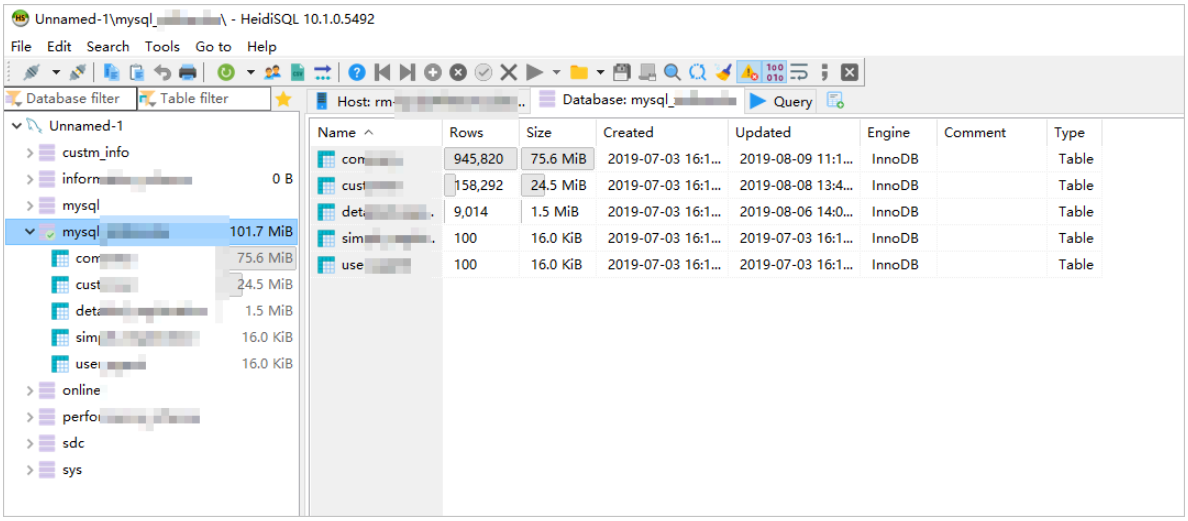
Parameter	Description
Hostname/IP	Enter the private or public IP address of the RDS instance. · If your database client is deployed in an ECS instance that is in the same region and has the same network type as the RDS instance, you can use the private IP address of the RDS instance. For example, if the ECS and RDS instances are both in a VPC located in the China (Hangzhou) region, then you can use the private IP address of the RDS instance to create a secure, efficient connection. · In the other situations, use the public IP address of the the RDS instance. You can obtain the private and public IP addresses of the RDS instance by completing the following steps: a. Log on to the *RDS console*. b. In the upper-left corner of the page, select the region where the RDS instance is located. c. Find the RDS instance and click its ID. d. On the displayed Basic Information page, find the private and public IP addresses and their corresponding port numbers. 
User	The username of the account that you use to access the RDS instance.
Password	The password of the account that you use to access the RDS instance.

Parameter	Description
Port	The port for the RDS instance to establish a connection. If you use the private IP address of the RDS instance to establish a connection, enter the private port number. If you use the public IP address of the RDS instance to establish a connection, enter the public port number.



4. Click Open.

   If the entered information is correct, the RDS instance can be connected.



Use the CLI to connect to an RDS instance

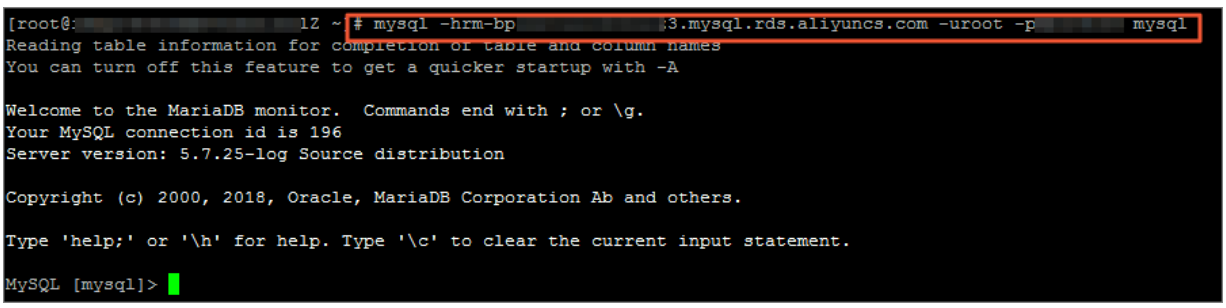
If MySQL is installed on your server, you can use the CLI to connect to an RDS instance as follows:

```
mysql -h<Host name> -P<Port number> -u<Username> -p<Password> -D<RDS
instance name>
```

Field	Description	Example
-h	The private or public IP address of the RDS instance. For more information, see #unique_44.	rm-bpxxxxxxxxxxxxxx.mysql.rds.aliyuncs.com
-P	The port for the RDS instance to establish a connection. · If you use the private IP address of the RDS instance to establish a connection, enter the private port number. · If you use the public IP address of the RDS instance to establish a connection, enter the public port number. **Note:** · The default port number is 3306. · If the port used by the RDS instance to establish a connection is Port 3306, you can retain the default value.	3306
-u	The username of the account that you use to access the RDS instance.	root

Field	Description	Example
-p	The password of the account that you use to access the RDS instance. **Note:** This field is optional. · If you do not enter the password in this field, the system prompts you to enter the password during subsequent operations. · If you enter the password in this field, note that no spaces are allowed between `-p` and the entered password.	`password233`
-D	The name of the RDS instance you want to access. **Note:** · This field is optional. · You can enter only the RDS instance name with `-D` removed.	`mysql`

Figure 3-1: Example of connecting to an RDS instance through CLI



3.6 What do I do if I cannot connect an ECS instance to an ApsaraDB for RDS instance?

This topic describes what you can do if you cannot connect an ECS instance to an RDS instance in various situations.

If you fail to connect ECS to RDS, one common reason is that the network type of the ECS instance differs from that of the RDS instance. Another common reason is that the IP address whitelist for the RDS instance does not contain the required IP addresses. The most common reasons and corresponding solutions are as follows:

ECS and RDS belong to different network types

The ECS instance runs in a VPC while the RDS instance runs in a classic network.

- **Solution 1 (recommended): Switch the RDS instance from its classic network to the VPC where the ECS instance resides. For detailed steps, see** *#unique_27*.

  Note:
  The RDS instance must run in the same VPC as the ECS instance after the switching so that they can communicate with each other through the intranet.
- **Solution 2: Purchase another ECS instance that runs in the classic network because ECS instances cannot be switched from a VPC to the classic network. A VPC is safer than the classic network. Therefore, we recommend that you use a VPC.**
- **Solution 3: Connect the ECS instance to the RDS instance through the Internet by using the public address of the RDS instance. This solution is inferior to solutions 1 and 2 in terms of performance, security, and stability.**

The ECS instance runs in the classic network while the RDS instance runs in a VPC.

- **Solution 1 (recommended): Switch the ECS instance from the classic network to the VPC where the RDS instance resides.**

  Note:
  The ECS instance must run in the same VPC as the RDS instance after the switching so that they can communicate with each other through the intranet.
- **Solution 2: Switch the RDS instance from its VPC to the classic network. However , a VPC is safer than the classic network. Therefore, we recommend that you use a VPC.**

- **Solution 3: Use the *ClassicLink* function. This function allows the ECS instances in the classic network to communicate with the resources in a VPC through the intranet.**
- **Solution 4: Connect the ECS instance to the RDS instance through the Internet by using the public address of the RDS instance. This solution is inferior to solutions 1, 2, and 3 in terms of performance, security, and stability.**

ECS and RDS are in different VPCs

Each VPC is a logically isolated network on Alibaba Cloud. If the ECS instance and RDS instance both run in VPCs, they must be in the same VPC so that they can communicate with each other through the intranet.

- **Solution 1 (recommended): Switch the RDS instance to the VPC where the ECS instance is located.**

  Specifically, switch the RDS instance from its VPC to the classic network and then switch from the classic network to the VPC where the ECS instance resides. For detailed steps, see *#unique_27*.
- **Solution 2: Establish an Express Connect channel between the two VPCs. For detailed steps, see *Interconnect two VPCs under the same account*.**
- **Solution 3: Connect the ECS instance to the RDS instance through the Internet. This solution is inferior to solutions 1 and 2 in terms of performance, security, and stability.**

ECS and RDS are in different regions

If the ECS instance is located in a region different from the RDS instance, they cannot communicate with each other through the intranet.

- **Solution 1: Release the ECS or RDS instance and purchase instances again.**
- **Solution 2: Set the network types of the ECS instance and RDS instance to VPCs, and establish an Express Connect channel between the two VPCs. For detailed steps, see *#unique_27* and *#unique_46*.**
- **Solution 3: Connect the ECS instance to the RDS instance through the Internet. This solution is inferior to solutions 1 and 2 in terms of performance, security, and stability.**

Incorrect IP address whitelist settings

- **The whitelist contains only the default IP address 127.0.0.1, which indicates that no devices are allowed to access the RDS instance. You need to add the IP address of the ECS instance to the whitelist. For detailed steps, see** *#unique_47*.
- **The IP address in the whitelist is 0.0.0.0. However, the correct format is 0.0.0.0/0.**

  Note:
  0.0.0.0/0 indicates that all devices are allowed to access the RDS instance. Please use it with caution.
- **The whitelist is set to the** *enhanced security mode*. **In this case, you need to check the following:**
  - **If you want the ECS instance to connect to the RDS instance through the VPC address, ensure that the private IP address of the ECS instance is added to the VPC whitelist of the RDS instance.**
  - **If you want the ECS instance to connect to the RDS instance through the classic network address, ensure that the private IP address of the ECS instance is added to the classic network whitelist of the RDS instance.**
  - **If you want the ECS instance to connect to the RDS instance through the Internet address, ensure that the public IP address of the ECS instance is added to the classic network whitelist of the RDS instance. The VPC whitelist does not restrict access from the Internet.**
- **The public IP address that you add to the whitelist is not the real outbound IP address of the ECS instance. Possible reasons are as follows:**
  - **The public IP address is not fixed and may change.**
  - **The IP address query tool or website may provide inaccurate IP addresses.**

  To find out the real IP address, see *Locate the real IP address*.

Domain name resolution failures

If your Domain Name Server (DNS) fails or its network interface card (NIC) configuration is changed, domain name resolution may fail. You can run the `ping` and `telnet` commands to check whether you can properly connect to the RDS instance.

```
ping <domain name>
```

```
telnet <domain name><port number>
```

Example:

If the communication is abnormal, you can modify the NIC configuration file of your DNS to resolve the problem by completing the following steps:

1. **Modify the NIC configuration file.**

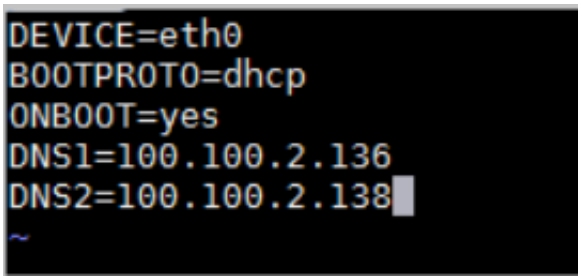
   ```
   vi /etc/sysconfig/network-scripts/<name of the NIC configuration
   file>
   ```

   Note:
   Fill the name of the NIC used by the ECS server in the <name of the NIC configuration file> field. You can run the `ifconfig` command to check the suffix. The default suffix is ifcfg-eth0.

2. **Add the following information to the end of the NIC configuration file:**

   ```
   DNS1=100.100.2.136
   DNS2=100.100.2.138
   ```

   Note:
   If the `DNS1` and `DNS2` parameters are set, you need to change their settings to the IP addresses shown above.

3. **Run the following command to restart your network service:**

   ```
   systemctl restart network
   ```

4. **Run the following command to check whether the modification is successful:**

   ```
   cat  /etc/resolv.conf
   ```

Common connection failures and solutions

Database type	Error message	Cause	Solution
MySQL or MariaDB TX	· ERROR 2003 (HY000): Can’t connect to MySQL server on ‘XXX’ (10038, 10060, or 110) · Cannot connect to the database: XXX	The network connection is abnormal.	*Click here.*
	· ERROR 1045 (HY000): #28000ip not in whitelist · ERROR 2801 (HY000): #RDS00ip not in whitelist, client ip is XXX	The IP address whitelist is set improperly.	*Click here.*
	· ERROR 1045 (28000): Access denied for user ‘XXX’@’XXX’ (using password: YES or NO)	The user name or password is incorrect.	*Click here.*
	· ERROR 2005 (HY000): Unknown MySQL server host ‘xxxxxxx’ (110 or 11004) · SQLSTATE[HY000] [2002] php_network_getaddresses: getaddrinfo failed: Name or service not known · Name or service not known	The DNS cannot parse IP addresses properly.	*Click here.*
SQL Server	Cannot connect to XXX. A network-related or instance-specific error occurs when a connection is being established with SQL Server. The server cannot be found or accessed. Check whether the instance name is correct. Also check whether the SQL Server is configured and allows remote access. (provider: TCP Provider, error: 0 - The receiver fails to respond correctly within the specified period or the host to be connected does not respond.) (Microsoft SQL Server, error: 10060 or 258)	The network connection is abnormal.	*Click here.*

Database type	Error message	Cause	Solution
	Cannot connect to XXX. A connection is established with the server, but an error occurs during the login. provider: TCP Provider, error: 0 - The specified network name is no longer available.)(Microsoft SQL Server, error : 64)	The IP address whitelist is set improperly.	*Click here.*
PostgreSQL /PPAS	Unable to connect to server: could not connect to server: Connection timed out (0x0000274C/10060)Is the server running on host “XXX.rds.aliyuncs.com” and acceptingTCP/IP connections on port XXX?	The network connection is abnormal.	*Click here.*
	· server closed the connection unexpectedly This probably means the server terminated abnormally before or while processing the request. · Error connecting to the server: FATAL : no pg_hba.conf entry	The IP address whitelist is set improperly.	*Click here.*

3.7 Read-only instances

3.7.1 Read-only instances overview

This topic provides an overview of read-only instances in RDS for MySQL. You can create one or more read-only instances to support a massive number of read requests and increase the throughput of applications.

Background information

An RDS for MySQL read-only instance is a read-only copy of the corresponding master instance. Changes to the master instance are automatically replicated to all

corresponding read-only instances. You can set the *replication delay* in each read-only instance.

Note:

- Currently the following MySQL versions and RDS editions support read-only instances:
  - MySQL 8.0 High-availability Edition (with local SSDs)
  - MySQL 5.7 High-availability Edition (with local SSDs)
  - MySQL 5.6
- Each read-only instance adopts a single-node architecture (that is, without slave nodes).

Pricing

Read-only instances are billed by using the pay-as-you-go billing method. For more information, see *ApsaraDB RDS for MySQL pricing*.

Features

Read-only instances in RDS for MySQL offer the following features:

- A read-only instance is located in the same or different zones from its corresponding master instance.
- The specifications of a read-only instance do not need to be the same as that of the corresponding master instance and can be changed at any time. We recommend that the specifications of a read-only instance be greater than or equal to that of the corresponding master instance. If the specifications of a read-only instance is lower than that of the corresponding master instance, the read-only instance may suffer from high delay or load.
- The *network type* of a read-only instance can be different from that of its corresponding master instance.
- No account or database maintenance is required for a read-only instance. Both accounts and databases are replicated from the corresponding master instance.
- When you create a read-only instance, the system automatically replicates the whitelists of the corresponding master instance. After a read-only instance is created, its whitelists are independent of the whitelists of the corresponding

master instance. If you want to change the whitelists of the read-only instance, see *Configure a whitelist for an RDS for MySQL instance*.

- Read-only instances support monitoring and alerting. Specifically, up to 20 performance metrics, including the disk capacity, IOPS, number of connections, CPU utilization, and network traffic.

Limits

- The number of read-only instances that a master instance can have varies depending on its memory size.

Database type	Memory	Max number of read-only instances
MySQL	⩾ 64 GB	10
	< 64 GB	5

- Read-only instances do not support backup settings or manual backups.
- Read-only instances do not support the creation of temporary instances through backup files or backups, or support the overwriting of instances by using backup sets.
- After a read-only instance is created, the master instance does not support data recovery through the direct overwriting of instances by using backup sets.
- You cannot migrate data to read-only instances.
- You cannot create or delete databases for read-only instances.
- You cannot create or delete accounts for read-only instances.
- You cannot authorize accounts or change account passwords for read-only instances.

FAQ

Can the accounts on the master instance be used on the read-only instances attached with it?

Accounts on the master instance are replicated to the read-only instances. You can use the accounts to read data from the read-only instances but cannot write data into the read-only instances.

3.7.2 Create an RDS for MySQL read-only instance

This topic describes how to create an RDS for MySQL read-only instance. You can create read-only instances to process massive read requests sent to the database

and increase the application throughput. A read-only instance is a read-only copy of the master instance. Changes to the master instance are also automatically synchronized to all relevant read-only instances.

For more information, see *Read-only instances overview*.

Prerequisites

The DB engine version and RDS edition are as follows:

- **MySQL 8.0 High-availability Edition (with local SSDs)**
- **MySQL 5.7 High-availability Edition (with local SSDs)**
- **MySQL 5.6**

Precautions

- **You can create read-only instances in your master RDS instance. However, you cannot switch an existing instance to a read-only instance.**
- **Creating a read-only instance does not affect your master RDS instance because the read-only instance replicates data from the slave instance.**
- **A read-only instance does not inherit the parameter settings of your master RDS instance. Instead, default parameter settings are generated. You can modify the parameter settings of a read-only instance on the RDS console.**
- **The quantity of read-only instances is as follows.**

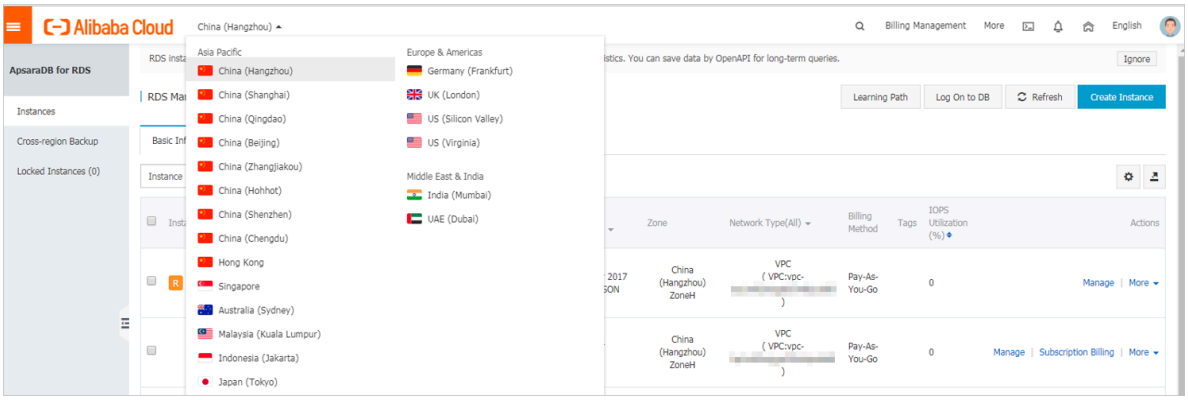
Database type	Memory	Max number of read-only instances
MySQL	⩾ 64 GB	10
	< 64 GB	5

- **A read-only instance is charged by using the pay-as-you-go billing method. Specifically, the fees for a read-only instance are deducted once per hour depending on the instance specifications. For more information, see the "Read-Only Instances" part at** *ApsaraDB RDS for MySQL pricing*.

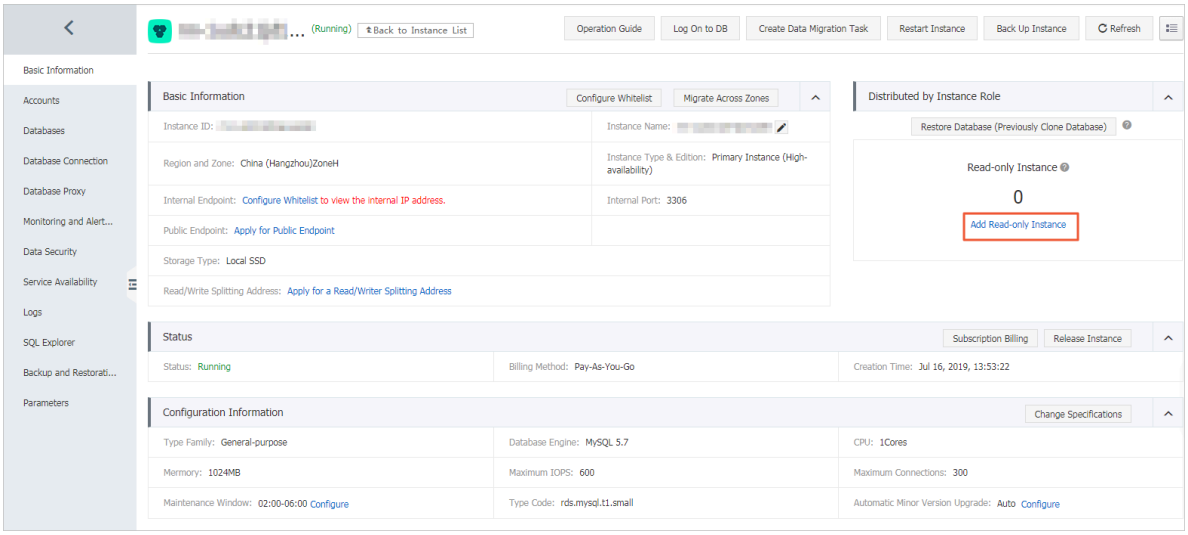
Create a read-only instance

1. **Log on to the** *RDS console*.

2. Select the target region.



3. Find the target RDS instance and click the instance ID.
4. Click Add Read-only Instance.



5. On the purchase page, choose the configuration of the read-only instance, and then click Buy Now.

Note:

- We recommend that the read-only instance and its corresponding master instance be located in the same VPC.
- To guarantee sufficient I/O for data synchronization, we recommend that the configuration of the read-only instance (specifically, the memory) is greater than or equal to that of the master instance.
- We recommend that you purchase multiple read-only instances based on your business needs to improve availability.

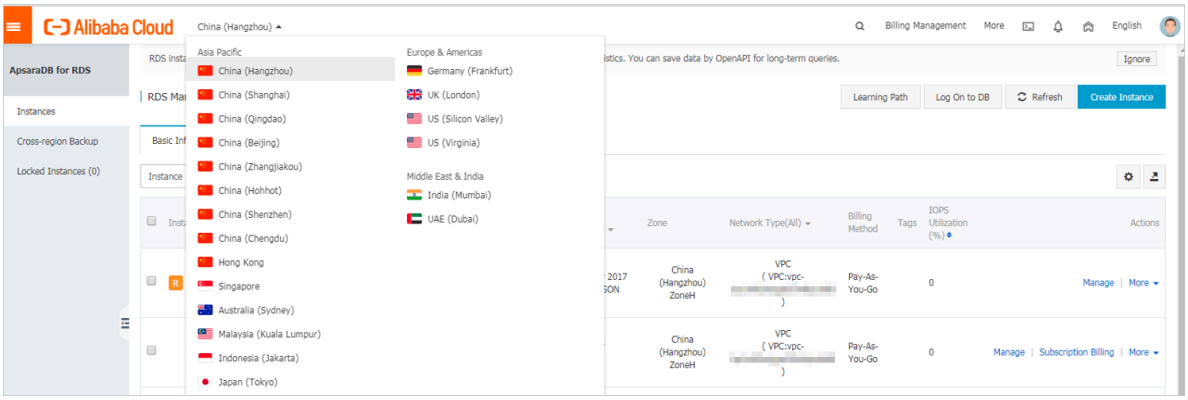
6. **On the Order Confirmation page, read and confirm you agree to Terms of Service, Service Level Agreement, and Terms of Use by selecting the checkbox, confirm the order details, and click Pay Now.**

   The instance creation takes a few minutes.

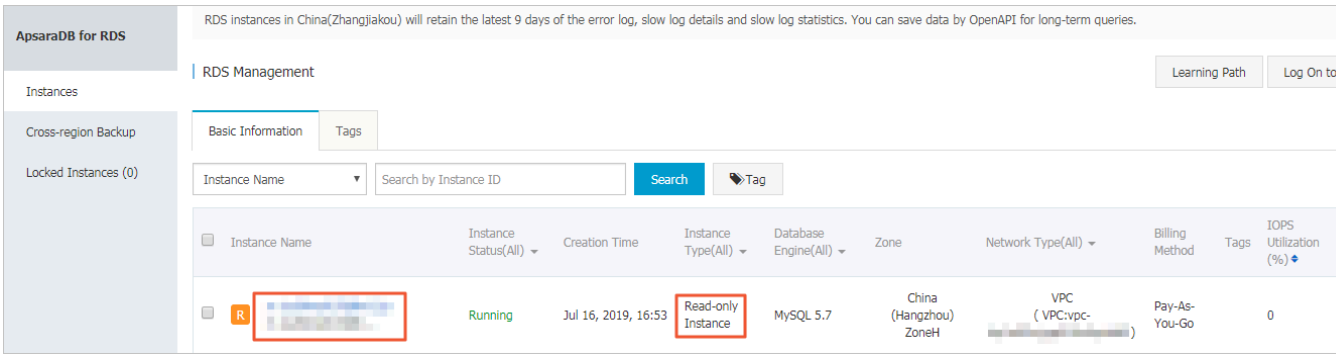
View a read-only instance

To view a read-only instance in the instance list, follow these steps:

1. **Log on to the *RDS console*.**
2. **Select the target region.**



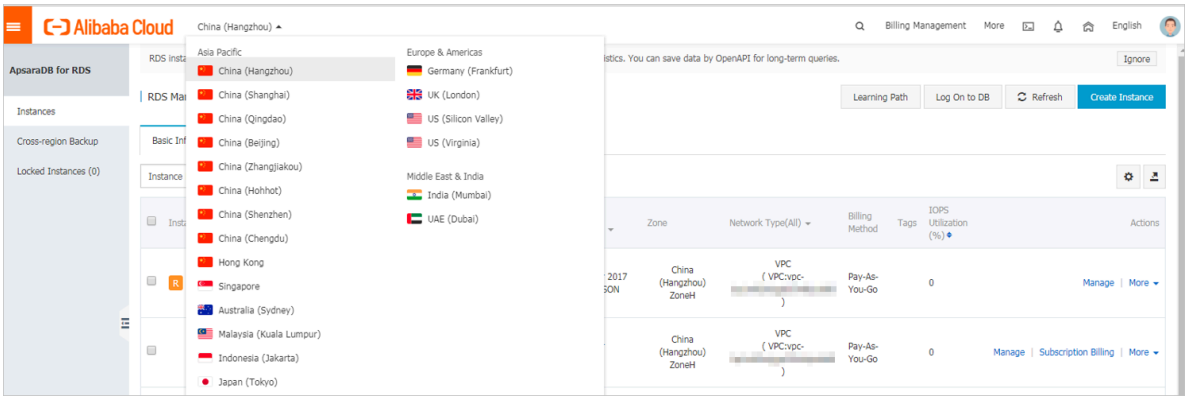
3. **Find the target read-only instance and click the instance ID.**



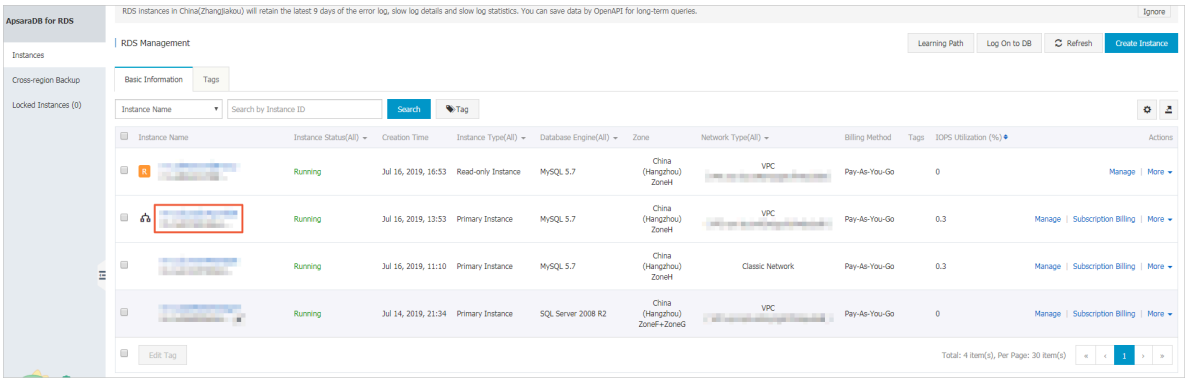
To view a read-only instance on the Basic Information page for the master instance, follow these steps:

1. **Log on to the *RDS console*.**

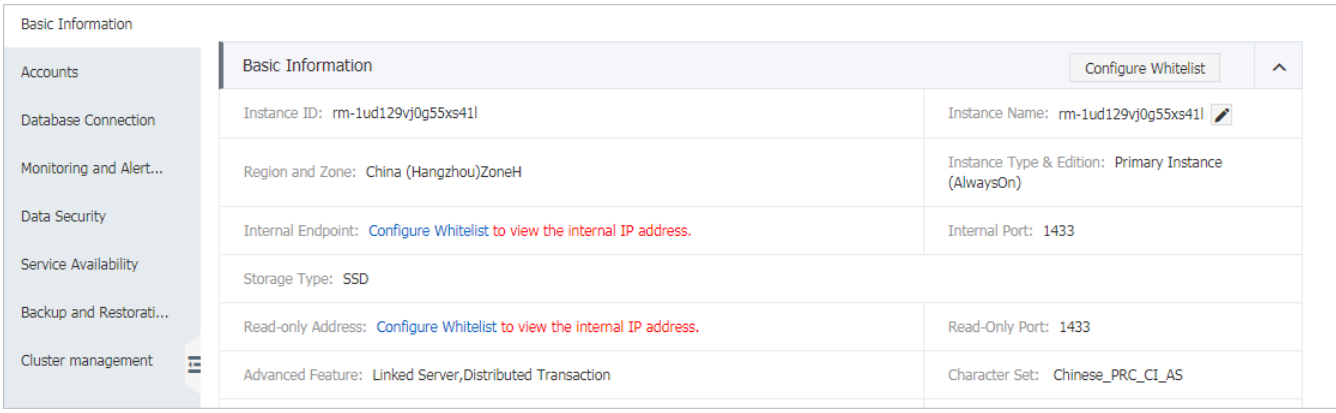
2. **Select the target region.**



3. **Find the master instance and click the instance ID.**

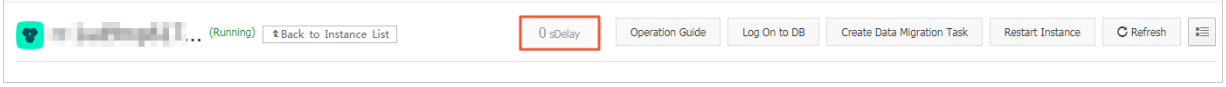


4. **On the Basic Information page, move the pointer over the number below Read-only Instance and click the ID of the read-only instance.**



View the delay time of a read-only instance

When a read-only instance replicates data from the master instance, the read-only instance may lag behind the master instance by a small amount of time. You can view the delay on the Basic Information page of the read-only instance.



APIs

API	Description
#unique_53	**Used to create an RDS read-only instance.**

3.8 Instance scale-out

4 Data migration and synchronization

4.1 Overview of RDS for MySQL data migration and synchronization

This topic provides an overview of RDS for MySQL data migration and synchronization.

Alibaba Cloud uses RDS to provide various data migration and synchronization solutions to meet different business requirements, such as migrating data to the cloud, migrating data between different cloud service providers, and synchronizing data. This allows you to migrate and synchronize databases to Alibaba Cloud RDS without affecting your business. Alibaba Cloud *DTS* provides the capabilities to migrate the schema, full data, and incremental data of MySQL databases and synchronize data between MySQL databases in a real-time manner. Additionally, you can migrate data from RDS for MySQL instances to on-premises databases by using physical and logical backup files.

Data migration

Scenario	Related documentation
Migrate data from an RDS instance to a POLARDB instance	• *Upgrade RDS for MySQL to POLARDB for MySQL with one click* • *Clone data from RDS for MySQL to POLARDB for MySQL with one click*
Migrate data from an on-premises database to an RDS for MySQL instance	• *Migrate data to an RDS for MySQL instance by using mysqldump* • *Migrate data from on-premises MySQL to RDS for MySQL* • *Migrate data from on-premises Oracle to RDS for MySQL*
Migrate data from a third-party cloud database to an Alibaba Cloud RDS for MySQL instance	• *#unique_60* • *Migrate a MySQL database from Google Cloud to Alibaba Cloud* • *#unique_62* • *#unique_63*

Scenario	Related documentation
Migrate data from an RDS instance to an on-premises MySQL database	*Migrate data from RDS for MySQL to an on-premises MySQL database*

4.2 Migrate data to an RDS for MySQL instance by using mysqldump

This topic describes how to migrate data to an RDS for MySQL instance by using mysqldump. mysqldump is easy to use but requires your database to go through a long service downtime. Therefore, we recommend that you use mysqldump when the data volume is small or a long service downtime is allowed.

Background information

ApsaraDB for RDS is fully compatible with native MySQL. Therefore, the process of migrating data from your database to an RDS for MySQL instance is similar to the process of migrating data from one MySQL server to another MySQL server.

Precautions

The data in the tables after migration is case-insensitive and centrally changed to lowercase letters.

Prerequisites

- **You have configured a whitelist, obtained a public endpoint, and created databases and accounts for the target RDS instance. For more information, see** *#unique_65* **.**
- **You have purchased an ECS instance.**

Procedure

1. **Create an account used to migrate data in your on-premises database.**

   ```
   CREATE USER'username'@'host' IDENTIFIED BY 'password';
   ```

   Parameter description:

   - `username`**: the username of the account.**
   - `host`**: the name of the host from which you log on to your on-premises database by using the account. As a local user, you can set this parameter to**

`localhost`. To log on from any host, you can set this parameter to the wildcard `%`.

- `password`: the password of the account.

For example, if you want to create an account whose username and password are `William` and `Changme123`, respectively, and enable the account to log on to your on-premises database from any host, then run the following command:

```
CREATE USER'William'@'%' IDENTIFIED BY 'Changme123';
```

2. Grant permissions to the account in your on-premises database.

```
GRANT SELECT ON databasename.tablename TO 'username'@'host' WITH
GRANT OPTION;
GRANT REPLICATION SLAVE ON databasename.tablename TO 'username'@'
host' WITH GRANT OPTION;
```

Parameter description:

- `privileges`: the permissions of the account, such as `SELECT`, `INSERT`, and `UPDATE`. To grant all permissions to the account, set this parameter to `ALL`.
- `databasename`: the name of your on-premises database. To grant the permissions for all databases to the account, set this parameter to the wildcard `*`.
- `tablename`: the name of a table. To grant the permissions for all tables to the account, set this parameter to the wildcard `*`.
- `username`: the username of the account.
- `host`: the name of the host from which you log on to your on-premises database by using the account. As a local user, you can set this parameter to

`localhost`. To log on from any host, you can set this parameter to the wildcard `%`.

- `WITH GRANT OPTION`: an optional parameter, which is used to grant the account the permission to run the `GRANT` command.

For example, If you want to grant all permissions for all databases and tables to the account `William` and allow the account to log on from any host, then run the following command:

```
GRANT ALL ON*. * TO 'William'@'%';
```

3. Use mysqldump to export data from your on-premises database as a data file.

Note:

Do not update the data during the data export. This step exports data only, excluding stored procedures, triggers, and functions.

```
mysqldump -h localIp -u userName -p --opt --default-character-set=
utf8 --hex-blob dbName --skip-triggers > /tmp/dbName.sql
```

Parameter description:

- `localIp`: the IP address of the server where your on-premises database is located.
- `userName`: the username of the account used to migrate data from your on-premises database.
- `dbName`: the name of your on-premises database.
- `/tmp/dbName.sql`: the name of the generated backup file.

4. Use mysqldump to export stored procedures, triggers, and functions.

Note:

If no stored procedures, triggers, or functions are used in the database, you can skip this step. When exporting stored procedures, triggers, and functions, you must remove the `DEFINER` parameter to guarantee compatibility with RDS.

```
mysqldump -h localIp -u userName -p --opt --default-character-set=
utf8 --hex-blob dbName -R | sed -e 's/DEFINER[ ]*=[ ]*[^*]*\*/\*/'
 > /tmp/triggerProcedure.sql
```

Parameter description:

- `localIp`: **the IP address of the server where your on-premises database is located.**
- `userName`: **the username of the account used to migrate data from your on-premises database.**
- `dbName`: **the name of your on-premises database you want to migrate.**
- `/tmp/triggerProcedure.sql`: **the name of the generated backup file.**

5. **Upload the data file and stored procedure file to your ECS instance.**

   In this example, assume that you want to upload the files to the following directories on your ECS instance:

   ```
   /tmp/dbName.sql
   /tmp/triggerProcedure.sql
   ```

6. **Log on to your ECS instance and import the data file and stored procedure file to the target RDS instance.**

   ```
   mysql -h intranet4example.mysql.rds.aliyuncs.com -u userName -p
   dbName < /tmp/dbName.sql
   mysql -h intranet4example.mysql.rds.aliyuncs.com -u userName -p
   dbName < /tmp/triggerProcedure.sql
   ```

   Parameter description:

   - `intranet4example.mysql.rds.aliyuncs.com`: **the endpoint of the target RDS instance. A private endpoint is used as an example.**
   - `userName`: **the username of the account used to migrate data to the RDS instance.**
   - `dbName`: **the name of your on-premises database you want to import.**
   - `/tmp/dbName.sql`: **the name of the data file you want to import.**
   - `/tmp/triggerProcedure.sql`: **the name of the stored procedure file you want to import.**

4.3 Migrate data from a third-party cloud database to ApsaraDB for RDS

4.3.1 Migrate a MySQL database from Google Cloud to Alibaba Cloud

This topic describes how to migrate a MySQL database from Google Cloud to Alibaba Cloud and the corresponding precautions.

Prerequisites

- **You have** *created an Alibaba Cloud RDS MySQL instance*.
- **You have** *created an account with read/write privileges*.

Limits

- **Structure migration does not support migration of events.**
- **For MySQL databases, DTS reads floating-point values (FLOAT and DOUBLE data types) with `round(column,precision)`. If the column definition does not specify the precision, the precision is 38 for FLOAT values and 308 for DOUBLE values.**
- **If the object name mapping function is used for an object, migration of objects relying on the object may fail.**
- **For incremental migration, you must enable binlog for the source MySQL instance.**
- **For incremental migration, binlog_format of the source database must be set to ROW.**

  Note:
  You can modify parameters of Google Cloud databases by choosingInstance details > Configuration > Edit configuration > Add database flags.
- **For incremental migration, if the source database version is MySQL 5.6 or later, binlog_row_image must be set to FULL.**
- **For incremental migration, if the source instance has binlog file ID disorder caused by cross-host migration, the incremental migration may have data loss.**

Precautions

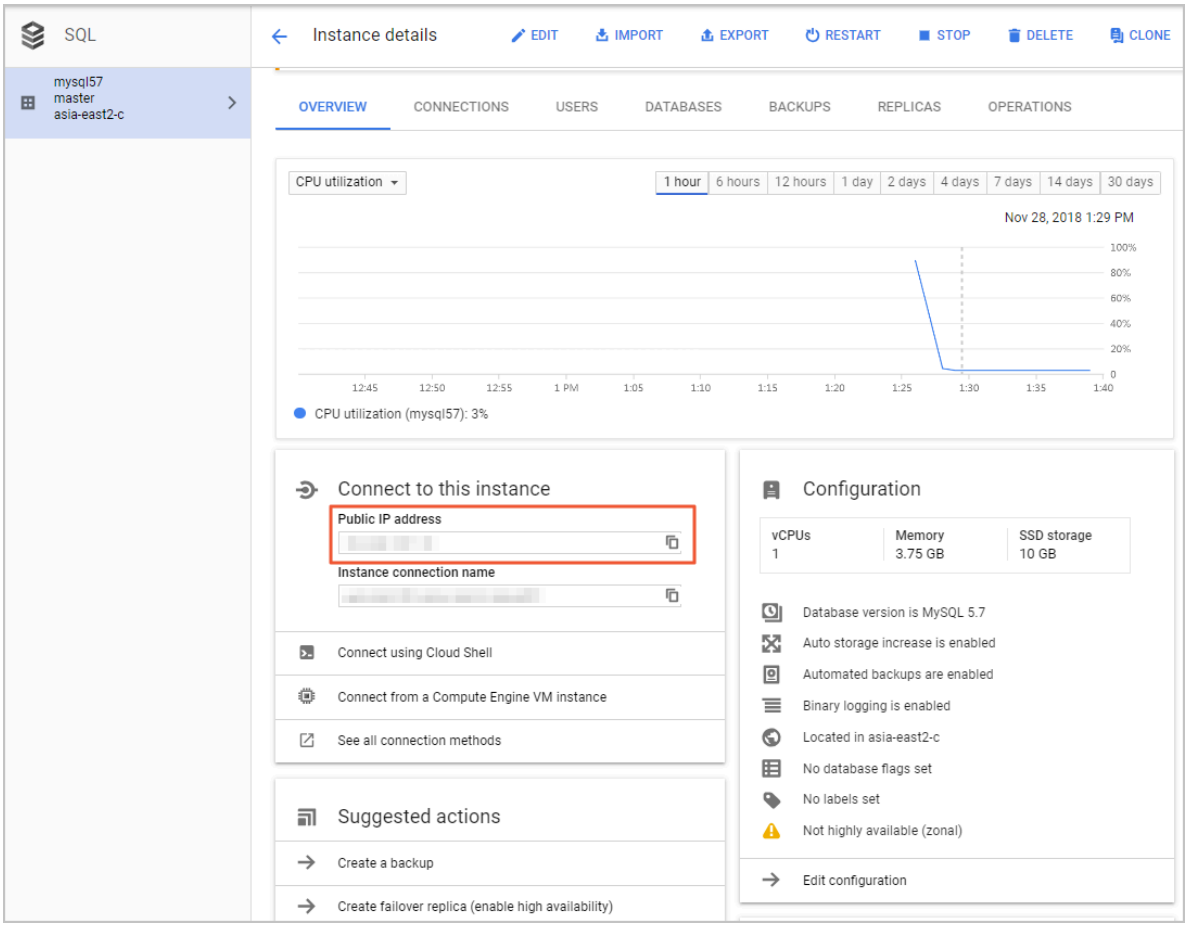
DTS automatically attempts to recover abnormal tasks of the past seven days. This may cause the new data in the target instance to be overwritten by the source

database data. Therefore, you must revoke the write permission of the DTS account that is used to access the target instance by running the `revoke` command.

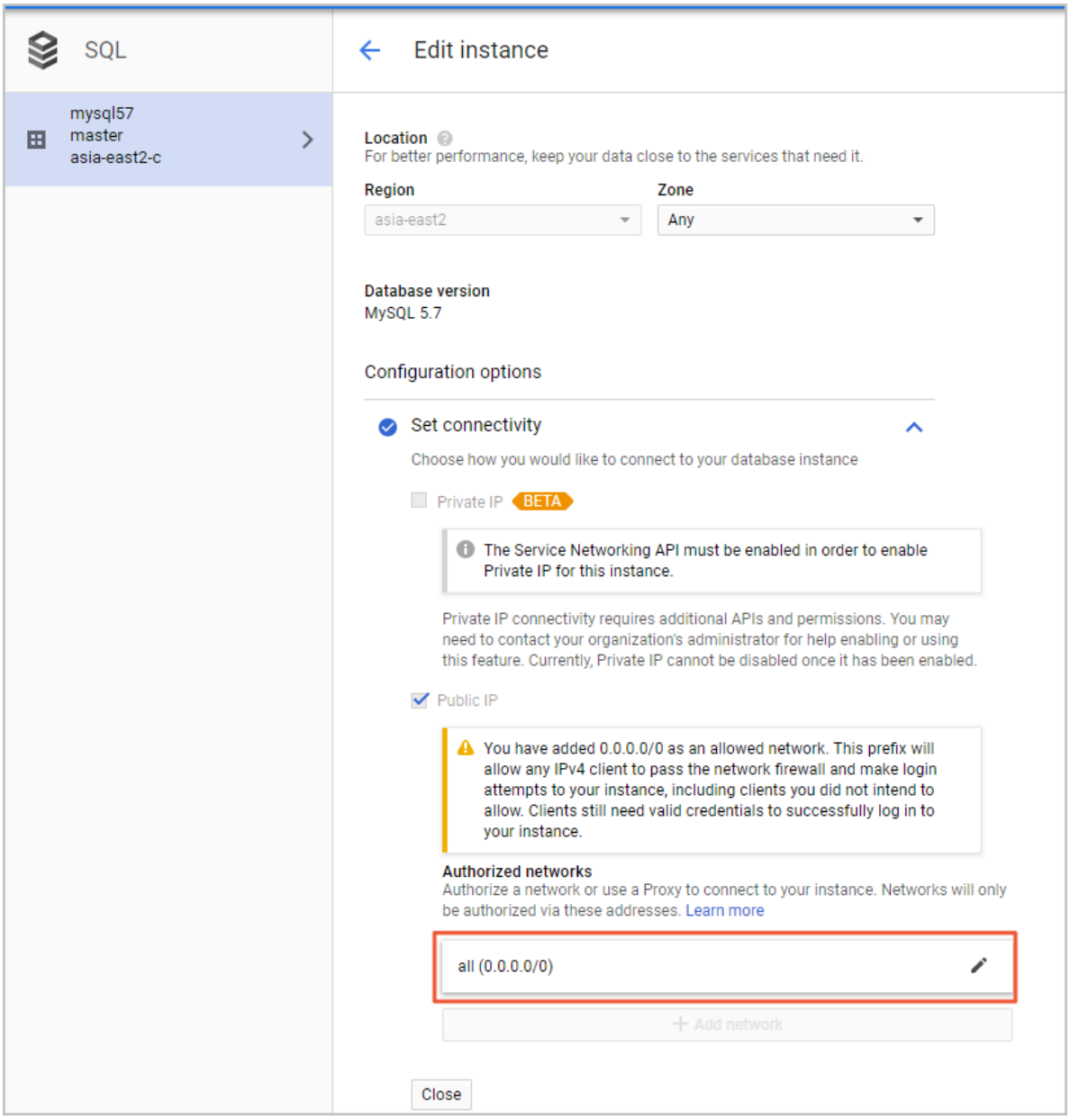
Procedure

1. **Log on to your database instance on Google Cloud. On the Instance details page, view `Public IP address`.**

> **Note:**
> **If a public IP address is not enabled, perform related settings by going to Configuration > Edit configuration > Set connectivity.**



2. **Choose Configuration > Edit configuration > Set connectivity > Add network, and then add the IP address of the *region of the source database instance* obtained from DTS.**

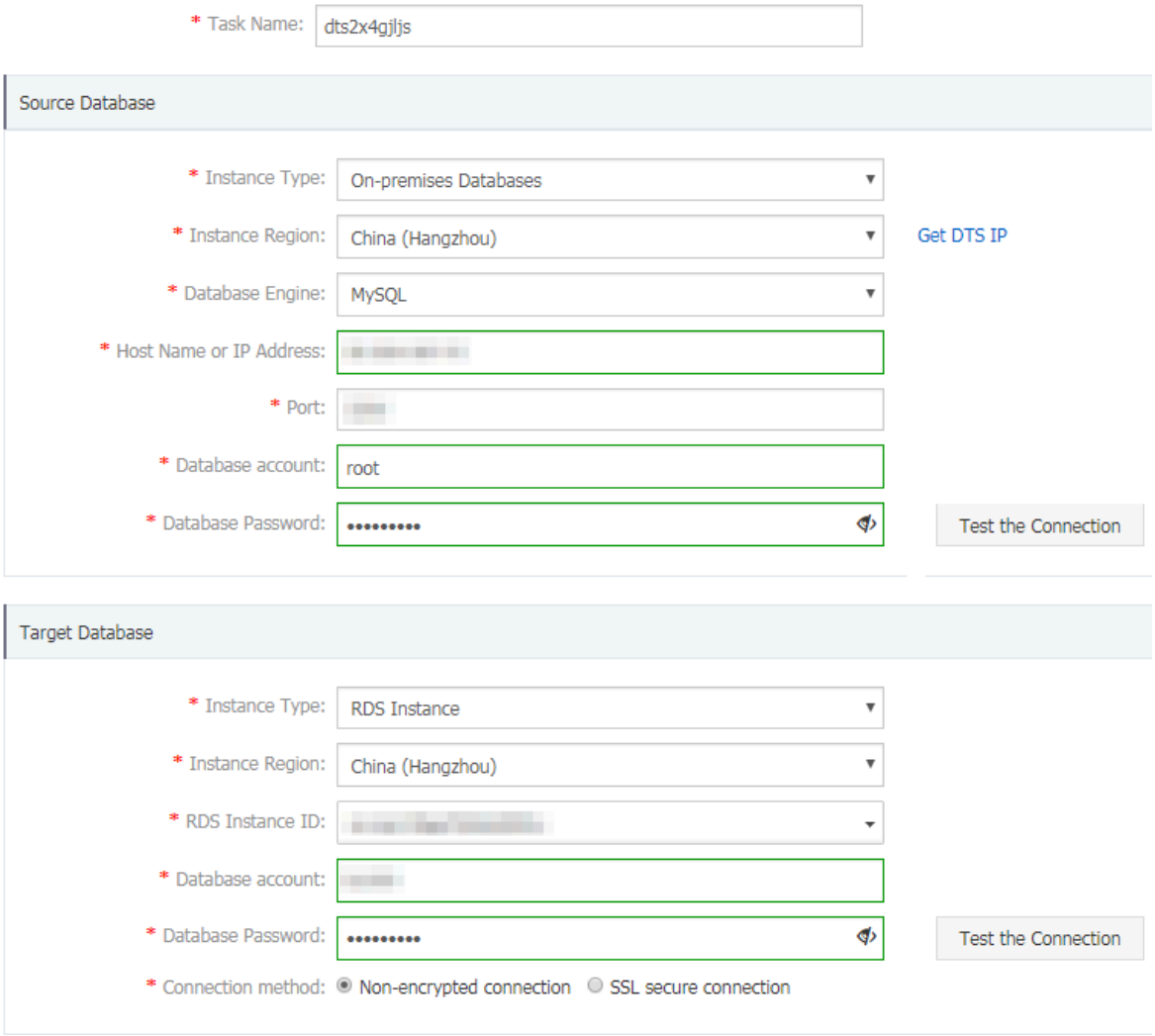


3. **Log on to the *DTS console*.**
4. **In the left-side navigation pane, click Data Migration. In the right pane, click Create Migration Task in the upper-right corner.**

5. Enter information about the source and target databases. The following table describes the parameters.

Database type	Parameter	Description
Source database (on Google Cloud)	Instance Type	Type of the instance in the source database. Select `On-premises Databases`.
	Instance Region	If you have configured access control for your instance, you must allow the specified Internet IP segment of the region to access the instance before configuring a migration task. **Note:** You can click Get DTS IP to view and copy the IP segment of the region.
	Database Engine	Source database type. Select `MySQL` .
	Host Name or IP Address	`Public IP address` of the database
	Port	Default port 3306
	Database account	Default superuser account `root`
	Database Password	Password of the root account
Target database (on Alibaba Cloud)	Instance Type	Type of the instance in the target database. Select `RDS Instance`.
	Instance Region	Region of the target instance
	RDS Instance ID	ID of the instance in the selected region. Select the ID of the target instance.
	Database account	An account with read and write permissions under the target instance
	Database Password	Account password

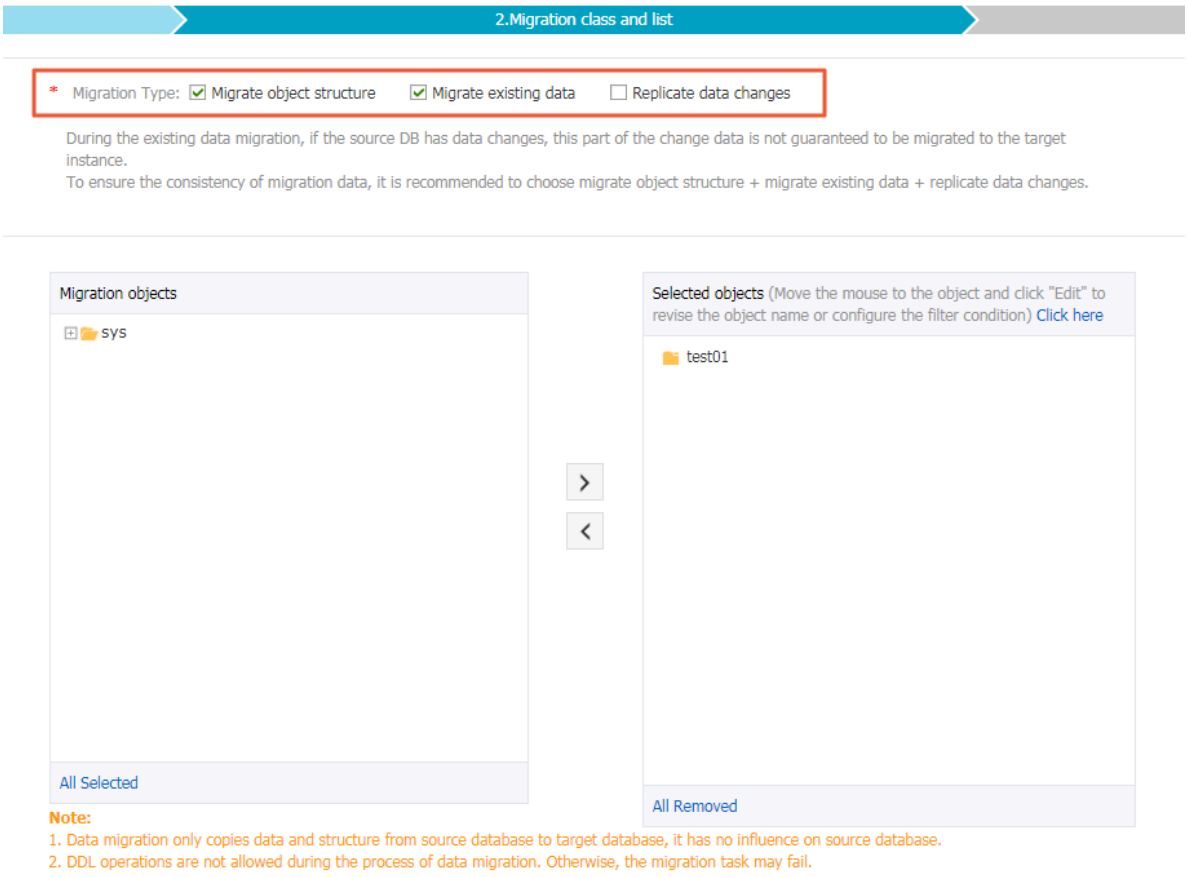
Database type	Parameter	Description
	Connection method	Select Non-encrypted connection or SSL secure connection. The latter greatly increases CPU consumption.



6. Click Test the Connection and confirm that the test results for both the source and target databases are `Test passed`.
7. Click Authorize Whitelist and Enter into Next Step.
8. Select the migration type. In the Migration objects area, select the target database and click to add the database to the Selected objects area.

Note:

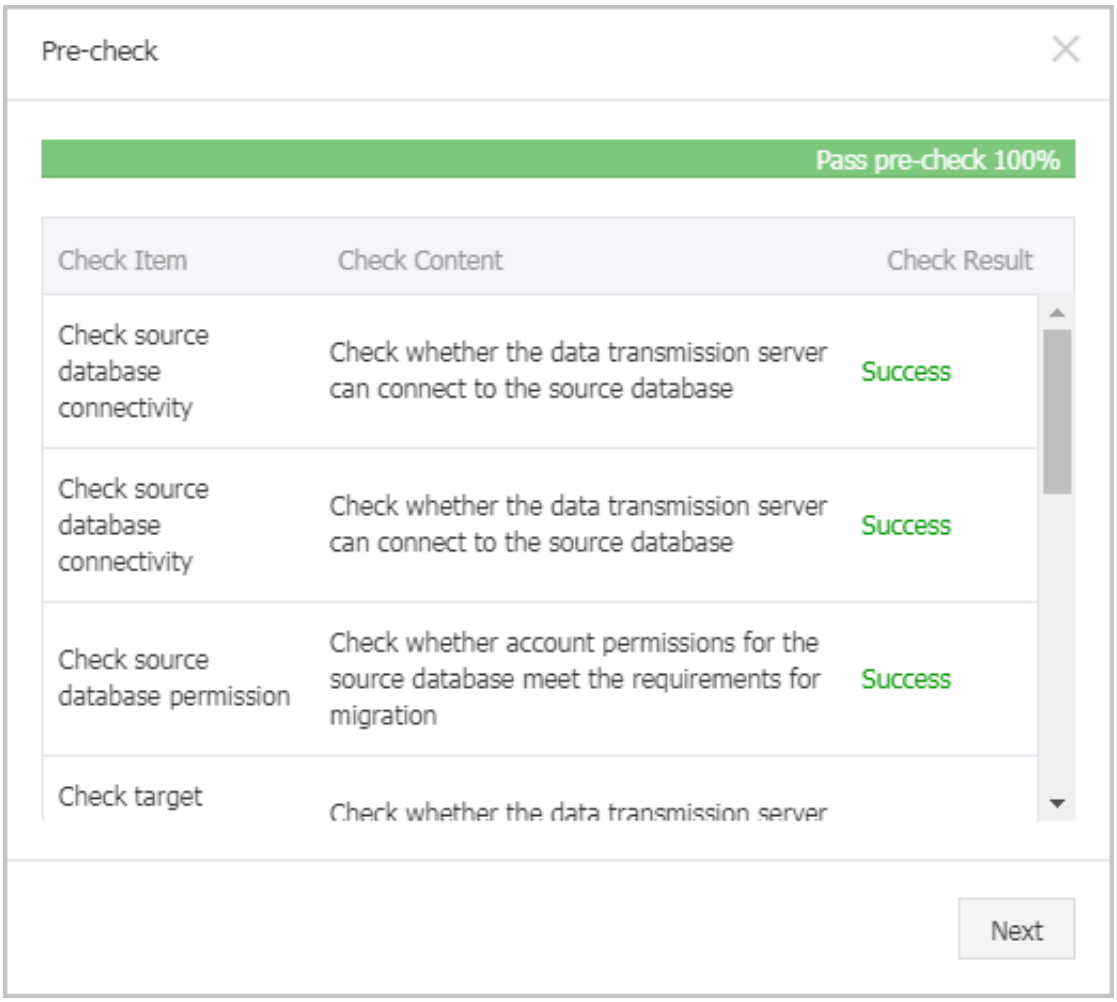
To maintain data consistency before and after migration, we recommend that you migrate the structure, full data, and incremental data.



9. **Click Pre-check and wait until the pre-check ends.**

Note:

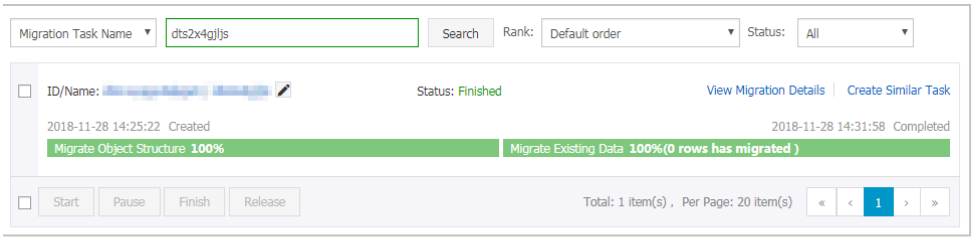
If the check fails, you can rectify faults according to error items and restart the task.



10. **Click Next. In the Confirm Purchase Configuration dialog box, read and select Service Terms of Data Transmission (Pay-As-You-Go), then click Buy and Start Now.**

Note:
Currently, structure migration and full migration are free of charge, while incremental migration is charged by the hour according to link specifications.

11. **Wait until the migration task is completed.**



4.3.2 Migrate MySQL data from AWS RDS to Alibaba Cloud RDS

This topic describes how to migrate MySQL data from AWS RDS to Alibaba Cloud RDS.

Background information

This topic describes how to use *Alibaba Cloud DTS* to migrate MySQL data from AWS RDS to Alibaba Cloud RDS.

Prerequisites

- **The source database instance to be migrated can access the Internet.**
- **You have created an RDS instance. For more information, see *Create an RDS instance*.**
- **You have created an account with read and write permissions. For more information, see *Create an account with read and write permissions*.**

Restrictions

- **Schema migration does not support the migration of events.**
- **DTS reads floating-point values (including float values and double values) in a column of the MySQL database by using the round(column,precision) method. If the value precision is not specified, the precision is 38 for float values and 308 for double values. Therefore, you must check whether the migration precision meets your service expectations.**
- **If object name mapping is enabled for an object, other objects depending on this object may fail to be migrated.**
- **If Incremental Data Migration is selected, binary logging must be enabled for the source MySQL instance.**
- **If Incremental Data Migration is selected, the binlog_format parameter of the source database must be set to row.**

- **If Incremental Data Migration is selected and the source MySQL version is 5.6 or later, the binlog_row_image parameter must be set to full.**
- **If binlog file ID disorder occurs in the source MySQL instance because of cross -host migration or reconstruction during incremental data migration, the incremental data being migrated may be lost.**

Precautions

- **We recommend that you back up data before performing migration tasks.**
- **DTS attempts to automatically recover abnormal tasks within seven days, which may lead to data in the source database overwriting the service data that has been written to the target database. Therefore, you must revoke the write permission of the DTS account that is used to access the target instance by running the `revoke` command.**

Procedure

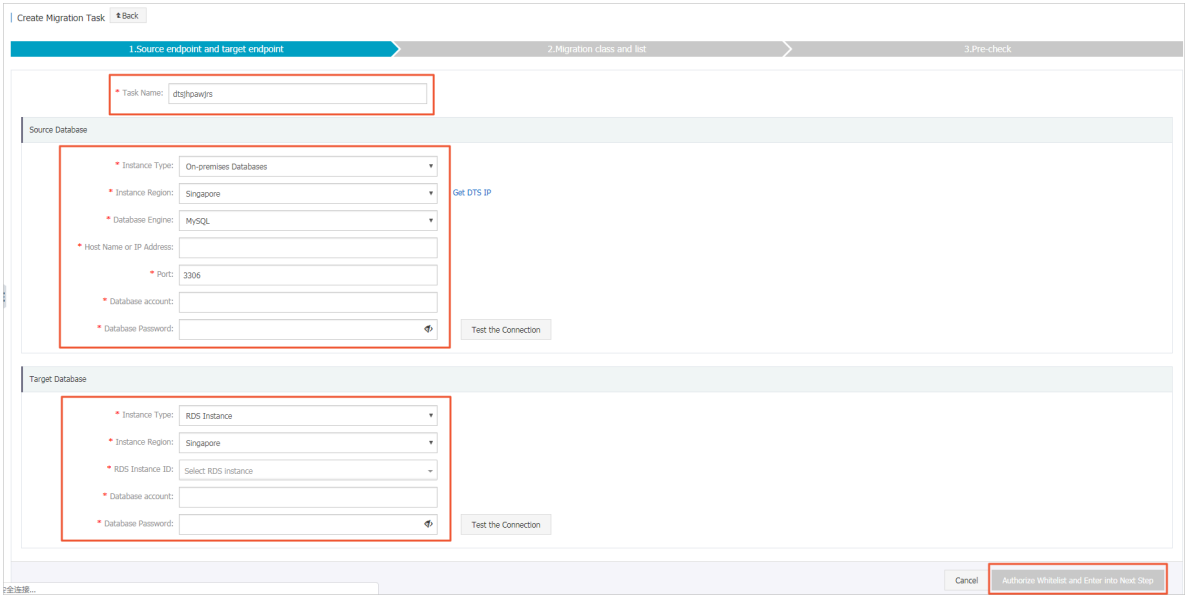
1. **Log on to the AWS MySQL database instance, and click Database Name. On the Connection page, view `Endpoint` and `Port`.**
2. **Log on to the *DTS console*.**
3. **In the left-side navigation pane, click Data Migration. In the right pane, click Create Migration Task in the upper-right corner.**
4. **Optional. Create a name for the task.**

   DTS generates a name for each task automatically. The task name is not required to be unique. You can modify the task name as required. We recommend that you choose an informative task name.
5. **Enter information about the source and target databases. The following table describes the parameters.**

Database type	Parameter	Description
Source database	Instance Type	The type of the source database instance. Select `User-Created Database with Public IP Address`.

Database type	Parameter	Description
	Instance Region	The region that the source database instance belongs to. If you have configured access control for your instance, you must allow the public IP address range of the specified region to access the instance before configuring a migration task. Note: You can click Get IP Address Segment of DTS to view and copy the IP address range of the region.
	Database Type	The source database type. Select `MySQL`.
	Hostname/IP Address	The `endpoint` of the AWS MySQL database.
	Port Number	The `port` of the AWS MySQL database.
	Database Account	An account with read and write permissions on the AWS MySQL database.
	Database Password	The password of the account for accessing the MySQL database.
Target database	Instance Type	Select `RDS Instance`.
	Instance Region	The region that the target database instance belongs to.
	RDS Instance ID	The ID of the target database instance in the selected region.
	Database Account	An account with read and write permissions on the target database instance.
	Database Password	The password of the account for accessing the target database instance.

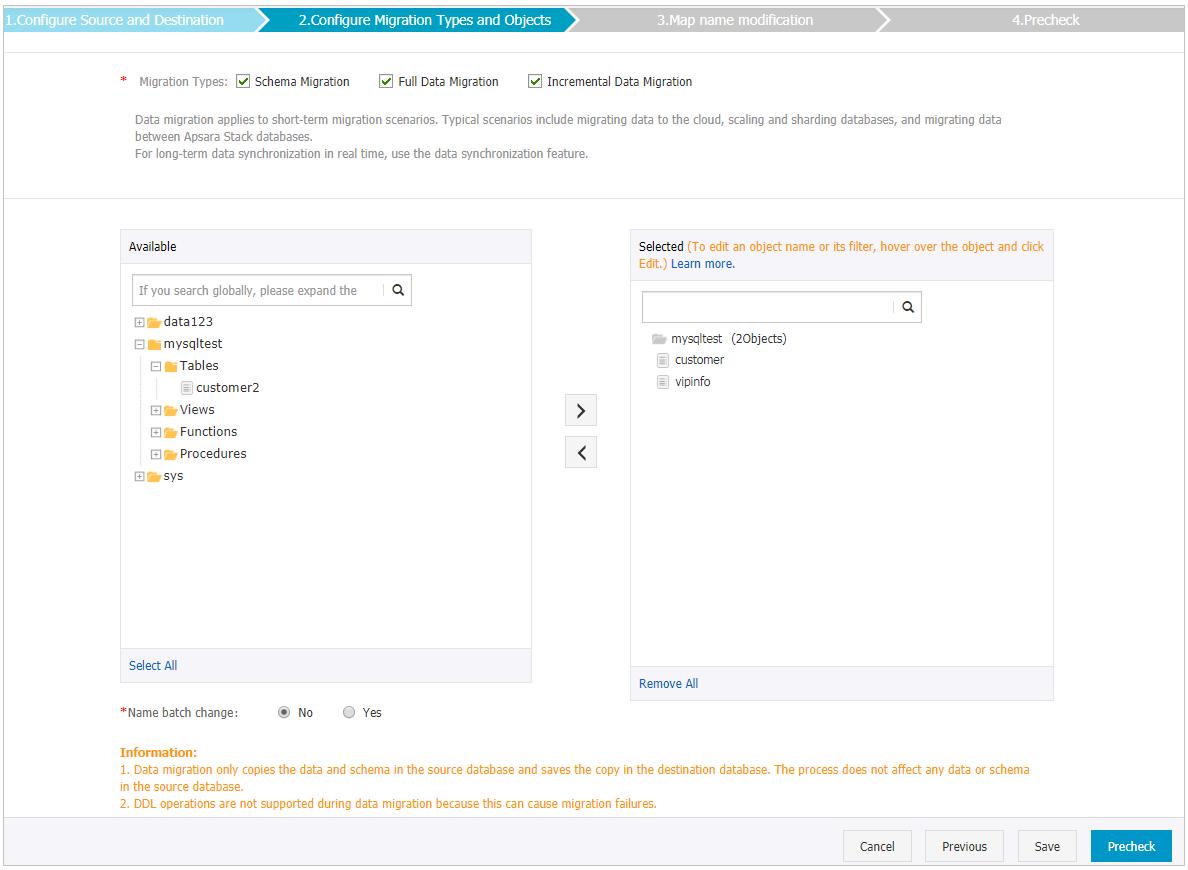
Database type	Parameter	Description
	Encryption	Select Non-encrypted or SSL-encrypted. The latter greatly increases CPU consumption.



6. Click Test the Connection and confirm that the test results for both the source and target databases are `Passed`.
7. Click Authorize Whitelist and Enter into Next Step in the lower right corner of the page.
8. Select a migration type. In the Available area, select the target database and click to add the database to the Selected area.

> **Note:**
> To maintain data consistency before and after migration, we recommend that you select Schema Migration, Full Data Migration, and Incremental Data Migration.

Currently, schema migration and full data migration are free of charge, while incremental data migration is charged by hour based on link specifications.



9. **Click Precheck and wait until the precheck ends.**

Note:

If the precheck fails, you can fix the problems as instructed and run the precheck again.

10. **Click Next. In the Confirm Settings dialog box that appears, read the term and agree to the Data Transmission Service (Pay-As-You-Go) Service Terms, and click Buy and Start.**

    If you select Incremental Data Migration, DTS synchronizes updates of the source database to the target RDS instance during the migration. The migration task does not end automatically. If you want to switch your service from AWS RDS to Alibaba Cloud RDS, we recommend that you use the following method. Pause data to write to the source database for a few minutes when incrementa l migration has no latency. Wait until the incremental migration enters non-latency status again. Stop the migration task and switch the service to the target RDS instance.

11. **Click the target region to view the migration status. The status will be Completed when the migration is completed.**

Note:

If no latency occurs during incremental data migration, the data on AWS and Alibaba Cloud RDS will be consistent and the migration task can be stopped.

AWS clears binlogs as quickly as possible, and incremental data migration depends on the binlogs of the source database. To prevent unsynchronized binlogs from being cleared, you can call the AWS RDS stored procedure `mysql.rds_set_configuration` to set the binlog retention period. For example, to prolong the binlog retention period to one day, you can run the following command to call the stored procedure:

```
call mysql.rds_set_configuration("binlog retention hours" 24)
```

Now you have migrated MySQL data from AWS RDS to Alibaba Cloud RDS.

4.4 Migrate data from a user-created database to ApsaraDB for RDS

This topic describes how to migrate data from a user-created database to ApsaraDB for RDS. By doing so, you can smoothly migrate services. The import method varies depending on the used DB engine.

Migrate data from an on-premises database to RDS for MySQL

- *Migrate data from on-premises MySQL to RDS for MySQL*
- *Migrate data from on-premises Oracle to RDS for MySQL*

4.5 Migrate data from RDS for MySQL to an on-premises MySQL database

This topic describes how to migrate data from RDS for MySQL to an on-premises MySQL database through a physical or logical backup file.

Migrate data by using a physical backup file

Background information

Due to software restrictions, you can restore data only in a Linux operating system . If you want to restore data to a Windows operating system, then you must first restore data to Linux and then migrate the data to Windows.

Prerequisites

Percona XtraBackup has been installed in your operating system. MySQL 5.6 and earlier versions require Percona XtraBackup 2.3. MySQL 5.7 requires Percona XtraBackup 2.4. You can download and install Percona XtraBackup from its official website. For more information, see *Percona XtraBackup 2.3* and *Percona XtraBackup 2.4* at the official website.

Procedure

This example assumes that your on-premises server runs the RHEL6/x64 operating system and the path for storing the physical backup file is `/home/mysql/`.

1. **Download the physical backup file of the source RDS for MySQL instance and upload the file to the destination server. For more information about how to obtain the physical backup file, see *Download the data backup files and log backup files of an RDS for MySQL instance*. If the destination server can access the source RDS for MySQL instance, then you can run the `wget "url"` command to download the physical backup file. `url` in the command indicates the URL from which you can download the physical backup file.**
2. **Go to the path where the physical backup file is saved.**

   ```
   cd /home/mysql/
   ```

3. **Decompress the physical backup file.**

   ```
   tar vizxf filename.tar.gz
   ```

   ***filename.tar.gz* indicates the name of the physical backup file.**

4. **Check whether the databases contained in the physical backup file after compression are correct.**

   ```
    cd filename/ll
   ```

   The system displays the following information, where `db0dz1rv11f44yg2`, `mysql`, and `test` are the databases in the source RDS for MySQL instance:

   ```
   -rw-r--r-- 1 root root       269 Aug 19 18:15 backup-my.cnf
   drwxr-xr-x 2 root root      4096 Aug 21 10:31 db0dz1rv11f44yg2
   -rw-rw---- 1 root root 209715200 Aug  7 10:44 ibdata1
   ```

```
drwxr-xr-x 2 root root      4096 Aug 21 10:31 mysql
drwxr-xr-x 2 root root      4096 Aug 21 10:31 test
-rw-r--r-- 1 root root        10 Aug 19 18:15 xtrabackup_binary
-rw-r--r-- 1 root root        23 Aug 19 18:15 xtrabackup_binlog_info
-rw-r--r-- 1 root root        77 Aug 19 18:15 xtrabackup_checkpoints
-rw-r--r-- 1 root root      2560 Aug 19 18:15 xtrabackup_logfile
-rw-r--r-- 1 root root        72 Aug 19 18:15 xtrabackup_slave_info
```

5. **Recover the data file.**

```
 innobackupex --defaults-file=./backup-my.cnf --apply-log . /
```

When the system displays `innobackupex: completed OK!`, the data is restored.

6. **Modify the configuration file. Comment out `innodb_fast_checksum`, `innodb_page_size`, and `innodb_log_block_size` in the decompressed file *backup-my.cnf*, and add `datadir=/home/mysql` to the file.**

```
# This MySQL options file was generated by innobackupex-1.5.1.
# The MySQL Server
[mysqld]
innodb_data_file_path=ibdata1:200M:autoextend
innodb_log_files_in_group=2
innodb_log_file_size=524288000
#innodb_fast_checksum=0
#innodb_page_size=16364
#innodb_log_block_size=512
datadir=/home/mysql/
```

7. **Reinstall MySQL to obtain the root permissions.**

```
rm -rf mysql
mysql_install_db --user=mysql --datadir=/home/mysql/
```

When the system displays the following information, MySQL is reinstalled:

```
Installing MySQL system table...
OK
Filling help table...
```

```
OK
```

8. **Modify the file owner.**

```
chown -R mysql:mysql /home/mysql/
```

9. **Start the MySQL process.**

```
mysqld_safe --defaults-file=/home/mysql/backup-my.cnf &
```

10. **Log on to the database from a client.**

```
mysql -u root -p
```

11. **Verify the database integrity.**

```
show databases;
```

When the system displays the following information, the database is restored:

```
+--------------------+
| Database           |
+--------------------+
| information_schema |
| db0dz1rv11f44yg2   |
| mysql              |
| performance_schema |
| test               |
+--------------------+
```

Migrate data by using a logical backup file

This example assumes that your on-premises server runs the RHEL6/x64 operating system and the path for storing the logical backup file is */home/mysql/*.

Procedure

1. **Download the logical backup file of the source RDS for MySQL instance and upload the file to the destination server. For more information about how to obtain the logical backup file, see** *Download the data backup files and log backup files of an RDS for MySQL instance*. **If the destination server can access the source RDS for MySQL instance, you can run the** `wegt "url"` **command to download the logical backup file.** `url` **in the command indicates the URL from which you can download the logical backup file.**

2. **Go to the path where the logical backup file is saved.**

```
cd /home/mysql/
```

3. **Decompress the logical backup file.**

```
tar vizxf filename.tar.gz
```

filename.tar.gz **indicates the name of the logical backup file.**

4. **Decompress the compressed SQL file.**

```
 gunzip filename.sql.gz
```

filename.sql.gz **indicates the name of the compressed SQL file.**

5. **Import data to the destination database.**

```
 mysql -u userName -p -h hostName -P port dbName < filename.sql
```

filename.sql **indicates the name of the decompressed SQL file.**

4.6 Upgrade RDS for MySQL to POLARDB for MySQL with one click

ApsaraDB for POLARDB allows you to upgrade an RDS for MySQL instance to a POLARDB for MySQL cluster with one click.

ApsaraDB for POLARDB introduction

ApsaraDB for POLARDB is the next-generation relational cloud database developed by Alibaba Cloud, which has the following main advantages.

- **Large storage capacity: up to 100 TB of storage.**
- **High performance: up to 6x performance improvement over MySQL.**
- **Serverless storage: no need to purchase storage capacity in advance, which is automatically scaled and is billed by usage.**
- **Temporary upgrade: supports temporary upgrade of specifications to easily cope with short-term business peaks.**

For more information, see *#unique_70*.

Highlights

- **Free-of-charge**
- **Zero data loss during migration**

- **Incremental data migration is supported. The service interruption period is less than 10 minutes.**
- **Rollback is supported. The migration can be rolled back within 10 minutes after the migration fails.**

Migration process

1. *Migrate data from the source RDS instance*. **This operation creates an ApsaraDB for POLARDB cluster with the same data as that of the source RDS instance. The incremental data of the source RDS instance will be synchronized to the ApsaraDB for POLARDB cluster in real time.**

   > **Note:**
   > **You need to change the database connection point in applications to that of the ApsaraDB for POLARDB cluster, verify that services are running properly, and click Complete Migration within 7 days. If you click Complete Migration, data synchronization stops between the source RDS instance and the destination ApsaraDB for POLARDB cluster.**

2. **Click Switch. This operation sets the source RDS instance to the read-only mode and the destination ApsaraDB for POLARDB cluster to the read and write mode. The incremental data of the ApsaraDB for POLARDB cluster will be synchronized to the source RDS instance in real time. Modify the database connection point in applications.**

   For more information about the procedure, see *Switch to the new cluster*.

   > **Note:**
   > **After you switch to the new cluster, you can also** *roll back the migration*.

3. *Complete the migration*.

Precautions

- **Data migration can only be performed in the same region.**
- **The destination ApsaraDB for POLARDB cluster must contain information of the source RDS instance, including the account, database, IP address whitelist, and required parameters.**
- **The parameters of the source RDS instance cannot be modified during migration .**

Prerequisites

- **The source RDS instance is of the RDS for MySQL 5.6 high-availability version.**
- *Transparent Data Encryption (TDE)* **and** *Secure Sockets Layer (SSL)***are not enabled in the source RDS instance.**
- **The table storage engine of the source RDS instance is InnoDB.**

Migrate data from the source RDS instance

This operation creates an ApsaraDB for POLARDB cluster with the same data as that of the source RDS instance. The incremental data of the source RDS instance will be synchronized to the ApsaraDB for POLARDB cluster in real time.

1. **Log on to the** *ApsaraDB for POLARDB console*.
2. **Click Create Cluster.**
3. **Select Subscription or Pay-As-You-Go (Hourly Rate).**
4. **Set parameters listed in the following table.**

Parameter	Description
Region	The region where the source RDS for MySQL instance resides. **Note:** The new ApsaraDB for POLARDB cluster is also located in this region.
Create Type	The method of creating the cluster. Select Migrate from RDS. · Default Create Type: creates a new ApsaraDB for POLARDB cluster. · Clone from RDS: clones the data of the selected RDS instance to an ApsaraDB POLARDB cluster. · Migration from RDS: clones the data of the selected RDS instance to an ApsaraDB for POLARDB cluster and keeps the data synchronized between the RDS instance and the ApsaraDB for POLARDB cluster. The binlogging feature is enabled for the new cluster by default.
RDS Engine Type	The engine type of the source RDS instance, which cannot be changed.
RDS Engine Version	The engine version of the source RDS instance, which cannot be changed.

Parameter	Description
Source RDS instance	The source RDS instances for selection, which do not include read-only instances.
Primary availability zone	The zone of the instance. A zone is an independent physical area located within a region. There are no substantive differences between the zones. You can deploy the ApsaraDB for POLARDB cluster and the ECS instance in the same zone or in different zones.
Network Type	The network type of the ApsaraDB for POLARDB cluster, which cannot be changed.
VPC Vswitch	The VPC and VSwitch to which the ApsaraDB for POLARDB cluster belongs. Make sure that you place your ApsaraDB for POLARDB cluster and the ECS instance to be connected in the same VPC. Otherwise, they cannot communicate with each other through the internal network to achieve optimal performance.
Database Engine	The database engine of the ApsaraDB for POLARDB cluster, which cannot be changed.
Node Specification	The node specifications of the ApsaraDB for POLARDB cluster. Select the specifications as required. We recommend that you select specifications that are at least the same as those of the source RDS instance. All ApsaraDB for POLARDB nodes are dedicated ones with stable and reliable performance. For more information, see *#unique_77*.
Number Nodes	The number of nodes. You do not need to specify this parameter. The system will create a read-only node with the same specifications as those of the primary node by default.
Storage Cost	The storage capacity. You do not need to specify this parameter. The actual usage is billed hourly in pay-as-you-go mode. For more information, see *#unique_77*.
Cluster Name	The cluster name for business distinguishing. The system will automatically create a name for your ApsaraDB for POLARDB cluster if you leave it blank. You can also modify the name after the cluster is created.

5. Specify Duration (only applicable to subscription clusters), and click Buy Now on the right side of the page.

6. Confirm the order information, read the Service Agreement, select the checkbox to agree to it, and click Activate Now.
7. Log on to the *ApsaraDB for POLARDB console* and view the status of the new ApsaraDB for POLARDB cluster.

> **Note:**
>
> - After the cluster is created, it synchronizes data from the source RDS instance. You need to modify the database connection point in applications and click *Complete Migration* within 7 days. Otherwise, the data migration is canceled automatically.
> - You can also cancel the migration in this step. For more information about the impact, see *FAQ*.

Switch to the new cluster

Prerequisites

- *Data is migrated from the source RDS instance to the destination ApsaraDB for POLARDB cluster*.
- The value of Replication Delay is less than 60 seconds.

Procedure

After the prerequisites are met, you can switch to the destination ApsaraDB for POLARDB cluster, and change the database connection point in applications.

1. Log on to the *ApsaraDB for POLARDB console*.
2. Find the destination cluster and click the cluster ID.
3. On the Basics page, click Switch. In the dialog box that appears, click OK.

   This operation sets the source RDS instance to the read-only mode and the destination ApsaraDB for POLARDB cluster to the read and write mode. The incremental data of the ApsaraDB for POLARDB cluster will be synchronized to the source RDS instance in real time.

> **Note:**
>
> - You cannot switch to the new cluster if the replication delay exceeds 60 seconds.
> - The switch process generally takes less than 5 minutes.

4. **Refresh the page. When POLARDB Read/Write Status is Read and Write, change the database connection point in applications as soon as possible.**

> **Note:**
> **After you switch to the new cluster, you can also *roll back the migration*.**

Complete the migration

*Migrate data from the source RDS instance*After data is migrated from the source RDS instance to the destination ApsaraDB for POLARDB cluster, you need to change the database connection point in applications and click Complete Migration within 7 days. This operation stops data synchronization between the RDS instance and the ApsaraDB for POLARDB cluster.

> **Warning:**
> **This operation stops data synchronization between the RDS instance and the ApsaraDB for POLARDB cluster, and the *rollback* feature is no longer available. We recommend that you use the ApsaraDB for POLARDB cluster for a period of time to verify that it runs properly before clicking Complete Migration.**

1. **Log on to the *ApsaraDB for POLARDB console*.**
2. **Find the destination cluster and click the cluster ID.**
3. **On the Basics page, click Complete Migration. In the dialog box that appears, click OK.**

> **Note:**
> - **After you click OK, the system stops data synchronization within 2 minutes. During this period, the Complete Migration button will not disappear. Do not click it repeatedly.**
> - **You can choose whether to disable the binlogging feature for the ApsaraDB for POLARDB cluster. If this feature is disabled, the write performance can be improved slightly. However, you need to restart the ApsaraDB for POLARDB cluster.**

4. **Release the source RDS instance if it is not needed.**

Roll back the migration

After switching to the new cluster, you can also roll back the migration. By rolling back the migration, you restore the source RDS instance to the read and write mode and the destination ApsaraDB for POLARDB cluster to the read-only mode. Data of the source RDS instance will be synchronized to the destination ApsaraDB for POLARDB cluster. The procedure is as follows:

1. **Log on to the *ApsaraDB for POLARDB console*.**
2. **Find the destination cluster and click the cluster ID.**
3. **On the Basics page, click Rollback. In the dialog box that appears, click OK.**

> **Note:**
> **After you click OK, the source RDS instance enters the read and write mode and the destination ApsaraDB for POLARDB cluster enters the read-only mode. Data of the source RDS instance will be synchronized to the destination ApsaraDB for POLARDB cluster. When Source RDS Read/Write Status is Read and Write, change the database connection point in applications to that of the RDS instance as soon as possible.**

FAQ

- **Q: Will the source RDS instance be affected when data is migrated from the RDS instance?**

  A: No, the source RDS instance can run properly.
- **Q: Will smooth migration affect business?**

  A: Smooth migration ensures zero data loss during migration. The service interruption period is less than 10 minutes. You can roll back the migration if needed.
- **Q: What happens if I cancel the migration?**

  A: If the migration is canceled, you can modify the parameters of the source RDS instance. The ApsaraDB for POLARDB cluster returns to the read and write mode, but will not be released. When canceling the migration manually, you can choose whether to disable the binlogging feature for the ApsaraDB for POLARDB cluster. The binlogging feature is not disabled if the migration is automatically canceled.

4.7 Clone data from RDS for MySQL to POLARDB for MySQL with one click

ApsaraDB for POLARDB allows you to clone data from an RDS for MySQL instance to a new POLARDB for MySQL cluster with one click.

This feature creates a destination ApsaraDB for POLARDB cluster with the same data as that of the source RDS instance. The incremental data of the source RDS instance will not be synchronized to the destination ApsaraDB for POLARDB cluster.

> **Note:**
> **If you need to synchronize the incremental data of the source RDS instance to the destination ApsaraDB for POLARDB cluster in real time while the cluster is being created, that is, to smoothly migrate data without service interruption, see** *Upgrade RDS for MySQL to POLARDB for MySQL with one click*.

ApsaraDB for POLARDB introduction

ApsaraDB for POLARDB is the next-generation relational cloud database developed by Alibaba Cloud, which has the following main advantages.

- **Large storage capacity: up to 100 TB of storage.**
- **High performance: up to 6x performance improvement over MySQL.**
- **Serverless storage: no need to purchase storage capacity in advance, which is automatically scaled and is billed by usage.**
- **Temporary upgrade: supports temporary upgrade of specifications to easily cope with short-term business peaks.**

For more information, see *#unique_70*.

Highlights

- **Free-of-charge**
- **Zero data loss during cloning**

Precautions

- **Data cloning can only be performed in the same region.**
- **The destination ApsaraDB for POLARDB cluster must contain information of the source RDS instance, including the account, database, IP address whitelist, and required parameters.**

Prerequisites

- The source RDS instance is of the RDS for MySQL 5.6 high-availability version.
- *Transparent Data Encryption (TDE)* and *Secure Sockets Layer (SSL)*are not enabled in the source RDS instance.
- The table storage engine of the source RDS instance is InnoDB.

Procedure

1. Log on to the *ApsaraDB for POLARDB console*.
2. Click Create Cluster.
3. Select Subscription or Pay-As-You-Go (Hourly Rate).
4. Set parameters listed in the following table.

Parameter	Description
Region	The region where the source RDS for MySQL instance resides. Note: The destination ApsaraDB for POLARDB cluster is also located in this region.
Create Type	The method of creating the cluster. - Default Create Type: creates a new ApsaraDB for POLARDB cluster. - Clone from RDS: clones the data of the selected RDS instance to an ApsaraDB POLARDB cluster. - Migration from RDS: clones the data of the selected RDS instance to an ApsaraDB for POLARDB cluster and keeps the data synchronized between the RDS instance and the ApsaraDB for POLARDB cluster. The binlogging feature is enabled for the new cluster by default. Select Clone from RDS.
RDS Engine Type	The engine type of the source RDS instance, which cannot be changed.
RDS Engine Version	The engine version of the source RDS instance, which cannot be changed.
Source RDS instance	The source RDS instances for selection, which do not include read-only instances.

Parameter	Description
Primary availability zone	The zone of the instance. A zone is an independent physical area located within a region. There are no substantive differences between the zones. You can deploy the ApsaraDB for POLARDB cluster and the ECS instance in the same zone or in different zones.
Network Type	The network type of the ApsaraDB for POLARDB cluster, which cannot be changed.
VPC Vswitch	The VPC and VSwitch to which the ApsaraDB for POLARDB cluster belongs. Make sure that you place your ApsaraDB for POLARDB cluster and the ECS instance to be connected in the same VPC. Otherwise, they cannot communicate with each other through the internal network to achieve optimal performance.
Database Engine	The database engine of the ApsaraDB for POLARDB cluster, which cannot be changed.
Node Specification	The node specifications of the ApsaraDB for POLARDB cluster. Select the specifications as required. We recommend that you select specifications that are at least the same as those of the source RDS instance. All ApsaraDB for POLARDB nodes are dedicated ones with stable and reliable performance. For more information, see *#unique_77*.
Number Nodes	The number of nodes. You do not need to specify this parameter. The system will create a read-only node with the same specifications as those of the primary node by default.
Storage Cost	The storage capacity. You do not need to specify this parameter. The actual usage is billed hourly in pay-as-you-go mode. For more information, see *#unique_77*.
Cluster Name	The cluster name for business distinguishing. The system will automatically create a name for your ApsaraDB for POLARDB cluster if you leave it blank. You can also modify the name after the cluster is created.

5. Specify Duration (only applicable to subscription clusters) and click Buy Now on the right side of the page.
6. Confirm the order information, read the Service Agreement, select the checkbox to agree to it, and click Activate Now.

Next step

Change the database connection point in applications to that of the ApsaraDB for POLARDB cluster as soon as possible. For more information, see *#unique_79*.

FAQ

Q: Will the source RDS instance be affected when data is cloned from the RDS instance?

A: No, the source RDS instance can run properly.

5 Billing management

5.1 Switch from pay-as-you-go billing to subscription billing

This topic describes how to change the billing method of an RDS for MySQL instance from pay-as-you-go to (monthly or annual) subscription.

Impacts

Changing the billing method does not interrupt the running of your RDS instance.

Precautions

- **A read-only RDS instance is charged only by the pay-as-you-go billing method.**
- **You cannot change the billing method of an RDS instance from subscription to pay-as-you-go. To optimize your cost plan, you must evaluate your usage model thoroughly before you change the billing method of your RDS instance.**
- **If an RDS instance has an unpaid subscription order, the subscription order becomes invalid after you upgrade the instance type. In such case, you must first go to the *Orders* page in the RDS console to cancel the subscription order, and then change the billing method to subscription again.**

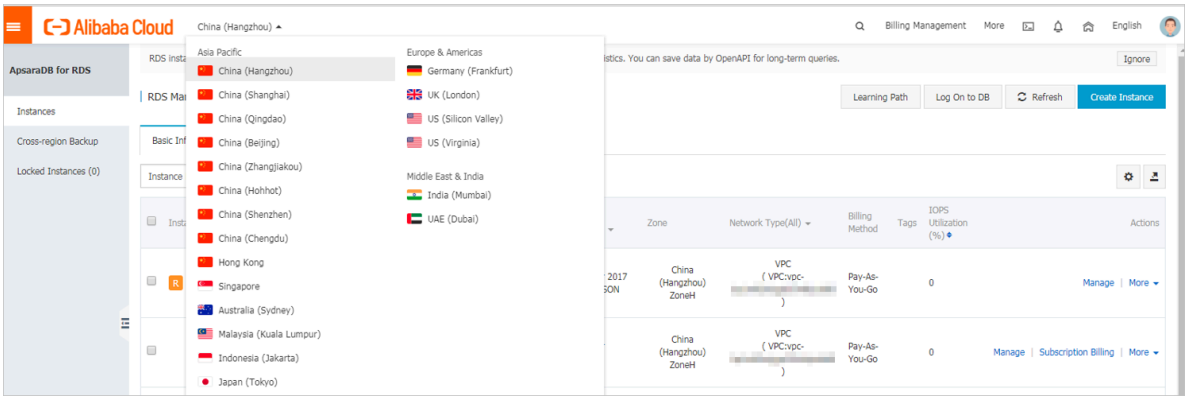
Prerequisites

- **The instance type cannot be a historical one, which means that the instance type must be available for sale. For more information about historical instance types, see *#unique_24*. Before you change the billing method of a historical-type RDS instance to subscription, you must change the instance type to one that is available for sale. For detailed steps, see *Change the configuration of an RDS for MySQL instance*.**
- **The RDS instance uses the pay-as-you-go billing method.**
- **The RDS instance is in the Running state.**
- **The RDS instance does not have an unpaid subscription order.**

Procedure

1. **Log on to the *RDS console*.**

2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and use one of the following two methods to open the Switch to Subscription Billing page.

   - In the Actions column, click Subscription Billing.
   - Click the instance ID. Then in the Status section of the Basic Information page, click Subscription Billing.

4. Select a duration of purchase.
5. Select Terms of Service, Service Level Agreement, and Terms of Use. Then click Pay Now.

> **Note:**
> The system generates a subscription order. If this order is not paid or canceled, you cannot change the billing method of this RDS instance from pay-as-you-go to subscription or purchase a new RDS instance. You can go to the *Orders* page to pay for or cancel this order.

6. Complete the payment.

5.2 Manually renew an RDS for MySQL instance

This topic describes how to manually renew an RDS for MySQL instance.

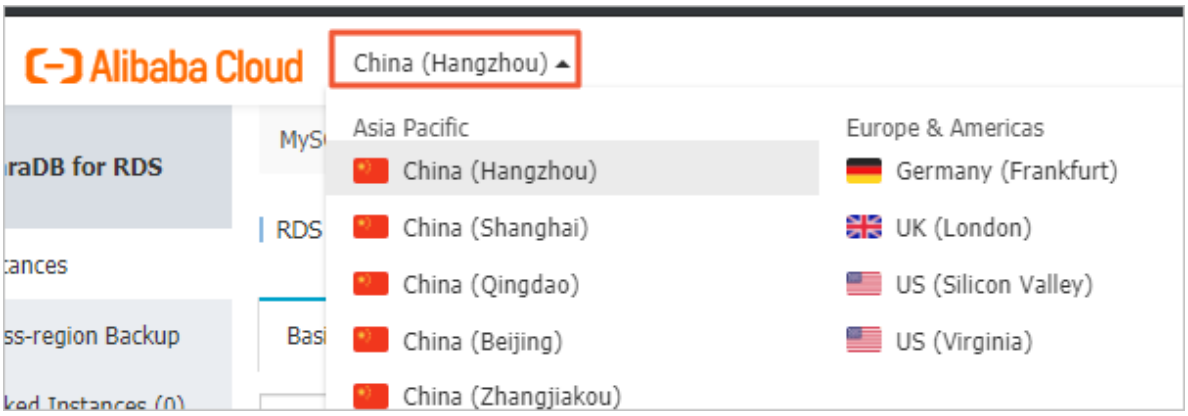
Each subscription-based instance has an expiration date. If an instance is not renewed in time before the instance expires, a service interruption or even data loss may occur. For more information about the impacts, see *Expiration and overdue policy*.

> **Note:**
> **A pay-as-you-go instance does not have an expiration date and no renewal is required.**

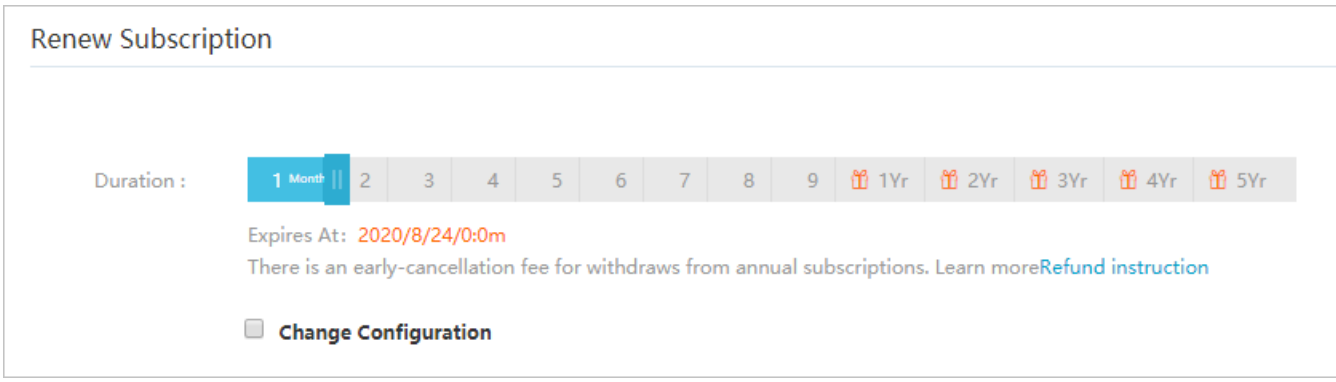
You can manually renew a subscription-based instance before it expires or within 15 days after it expires.

Method 1: Renew an RDS instance in the RDS console

1. **Log on to the *ApsaraDB for RDS console*.**
2. **In the upper-left corner of the page, select the region where the target RDS instance is located.**



3. **Find the target RDS instance and in the Actions column click Renew.**
4. **On the Renew Subscription page, select a duration. The longer the duration, the bigger discount you have.**

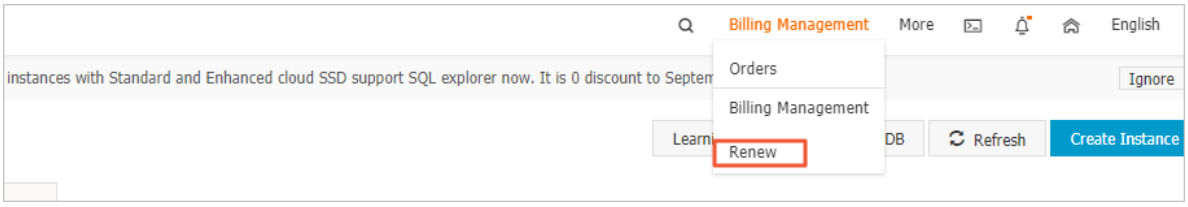


5. **Select Terms of Service, Service Level Agreement, and Terms of Use, and click Pay Now to complete the payment.**

Renew an RDS instance in the Renew console

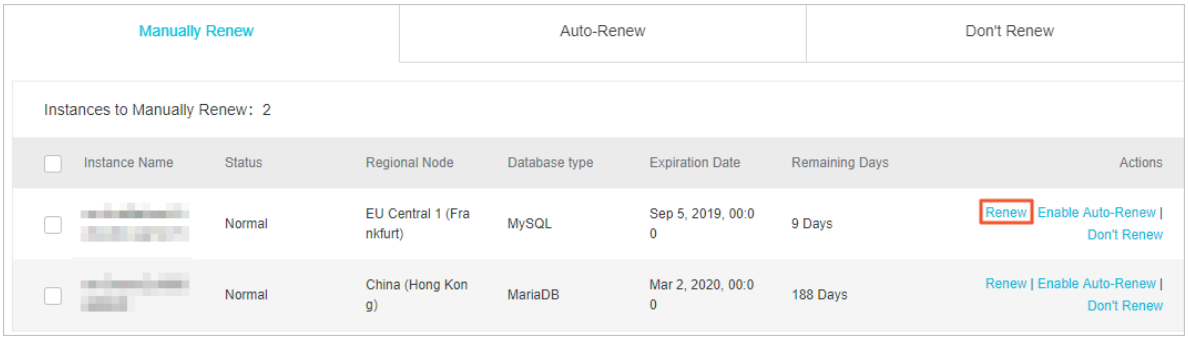
1. **Log on to the *ApsaraDB for RDS console*.**

2. In the upper-right corner of the page, choose Billing Management > Renew.



3. In the left-side navigation pane, click ApsaraDB for RDS.
4. On the Manually Renew tab, find the target RDS instance and in the Actions column click Renew

> **Note:**
>
> - If the target RDS instance is on the Don't Renew tab, you can click Enable Manual Renew in the Actions column to restore the instance to manual renewal.
> - If the target RDS instance is on the Auto-Renew tab, you can click Modify Auto-Renew in the Actions column, and then in the displayed dialog box select Disable Auto-Renew and click OK to restore the instance to manual renewal.



5. Select a duration, select the service agreement, and click Pay Now to complete the payment.

Auto-renewal

Enabling auto-renewal guarantees that your business runs smoothly without the need of manual renewal when your instance expires. For more information, see *Automatically renew an RDS for MySQL instance*.

5.3 Automatically renew an RDS for MySQL instance

This topic describes how to automatically renew an RDS for MySQL instance.

Each subscription-based instance has an expiration date. If an instance is not renewed in time when the instance expires, a service interruption or even data loss may occur. For more information about the impacts, see *Expiration and overdue policy*. Enabling auto-renewal guarantees that your business runs smoothly without the need of manual renewal when your instance expires.

> **Note:**
> **A pay-as-you-go-based instance does not have an expiration date and no renewal is required.**

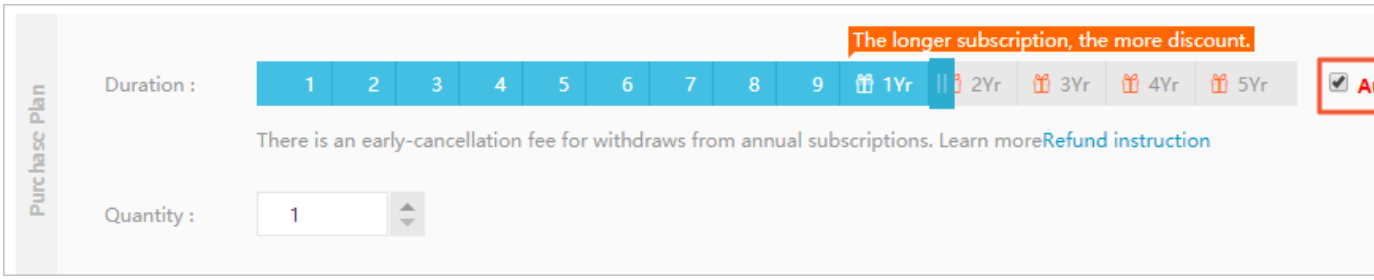
Precautions

- **If you have enabled automatic renewal for your subscription-based instance, a payment will be deducted three days before the expiration date. You can pay the fees by credit cards or coupons. Make sure that your credit card has sufficient balance.**
- **If you manually renew an instance before the automatic deduction date, the system will automatically renew the instance before the next expiration date.**
- **The automatic renewal function takes effect the next day after you enable it. If your instance expires the next day, renew it manually to prevent service interruption. For more information, see *Manually renew an RDS for MySQL instance*.**

Enable automatic renewal when you purchase an RDS instance

> **Note:**
> **After you enable automatic renewal, the system automatically renews your instance based on the specified Duration when the instance expires. For example, if you have purchased a three-month subscription-based instance and selected Auto-renewal, the fees are automatically paid every three months for each renewal.**

When you *purchase a subscription-based instance*, you can select Auto Renewal on the purchase page.

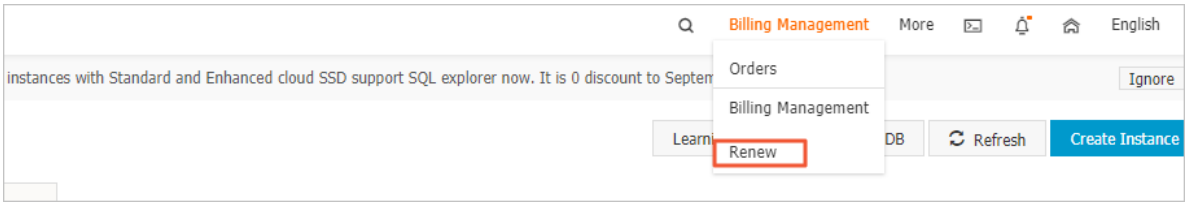


Enable automatic renewal after you purchase an RDS instance

Note:

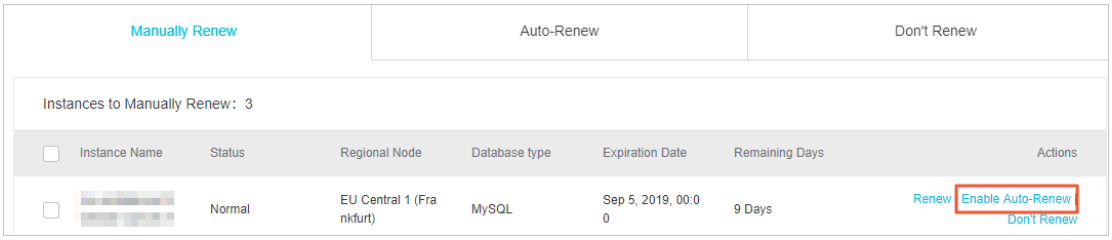
After you enable automatic renewal, the system automatically renews your instance based on the selected renewal duration. For example, if you select a three-month renewal duration, the fees are automatically paid every three months for each renewal.

1. **Log on to the *RDS console*.**
2. **In the upper-right corner, choose Billing Management > Renew.**



3. **In the left-side navigation pane, click ApsaraDB for RDS.**

4. On the Manually Renew or Auto-Renew tab, find the target RDS instance. You can enable automatic renewal for one or more RDS instances at a time.
   - Follow these steps to enable automatic renewal for one RDS instance:
     a. Find the target RDS instance and in the Actions column click Enable Auto-Renew.

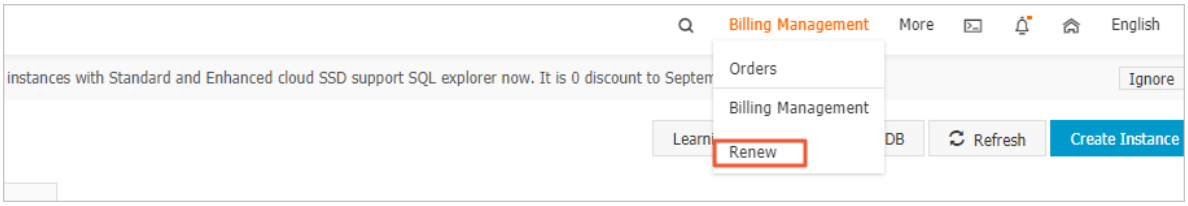


     b. In the displayed dialog box, set Auto-Renew Cycle and click Enable Auto-Renew.
   - Follow these steps to enable auto-renewal for more than one RDS instance:

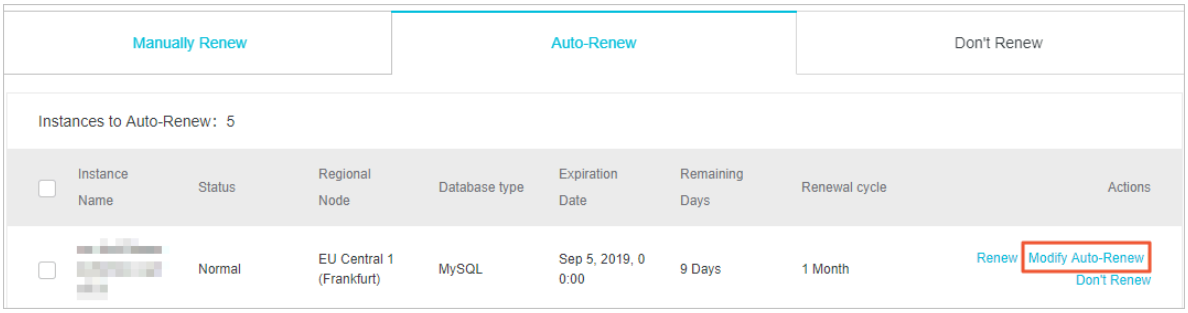
     Select the target RDS instances, and click Enable Auto-Renew below the instance list.
   - In the displayed dialog box, set Auto-Renew Cycle and click Enable Auto-Renew.

Change the auto-renew cycle of an RDS instance

1. Log on to the *RDS console*.
2. In the upper-right corner, choose Billing Management > Renew.



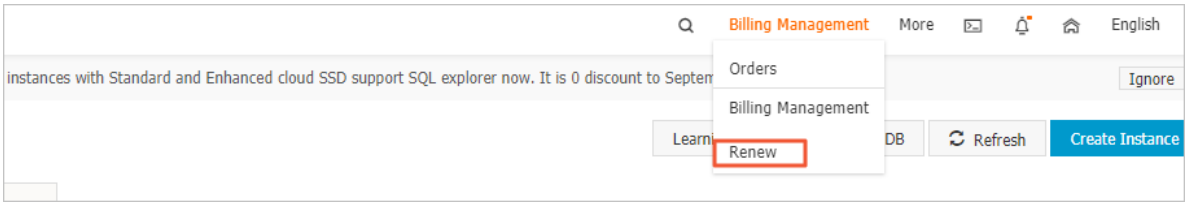
3. In the left-side navigation pane, click ApsaraDB for RDS.
4. On the Auto-Renew tab, find the target RDS instance and in the Actions column click Modify Auto-Renew.



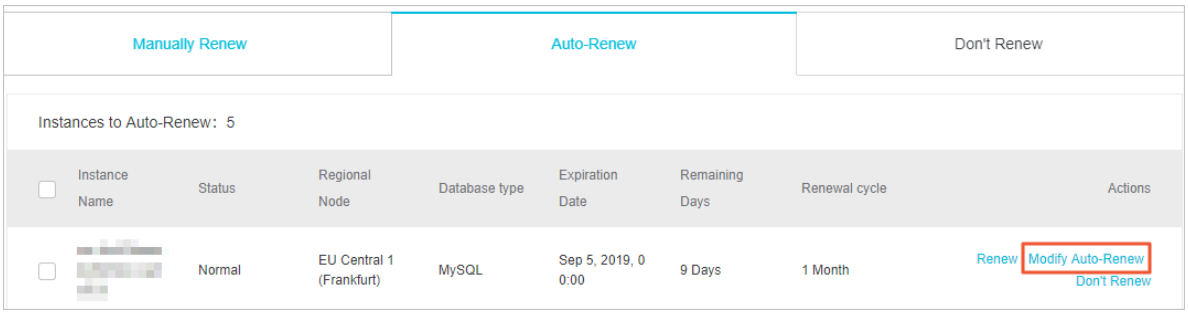
5. **In the displayed dialog box, select Modify Auto-Renew Cycle, select an auto-renew cycle, and click OK.**

Disable automatic renewal for an RDS instance

1. **Log on to the *RDS console*.**
2. **In the upper-right corner, choose Billing Management > Renew.**



3. **In the left-side navigation pane, click ApsaraDB for RDS.**
4. **On the Auto-Renew tab, find the target RDS instance and in the Actions column click Modify Auto-Renew.**



5. **In the displayed dialog box, select Disable Auto-Renew and click OK.**

APIs

Operation	Description
#unique_26	**Used to create an RDS instance.** **Note:** **Automatic renewal is enabled when you create the instance.**
#unique_85	**Used to renew a subscription-based RDS instance.** **Note:** **Automatic renewal is enabled after you create the instance.**

6 Instance management

6.1 Restart an RDS instance

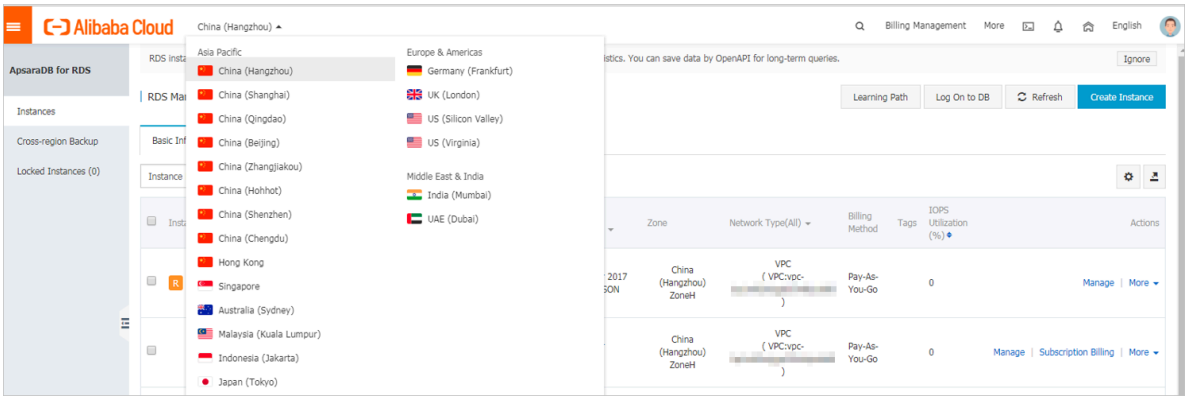
This topic describes how to manually restart an RDS instance in the RDS console if the number of connections exceeds the threshold or any performance issue occurs for the instance.

Impact

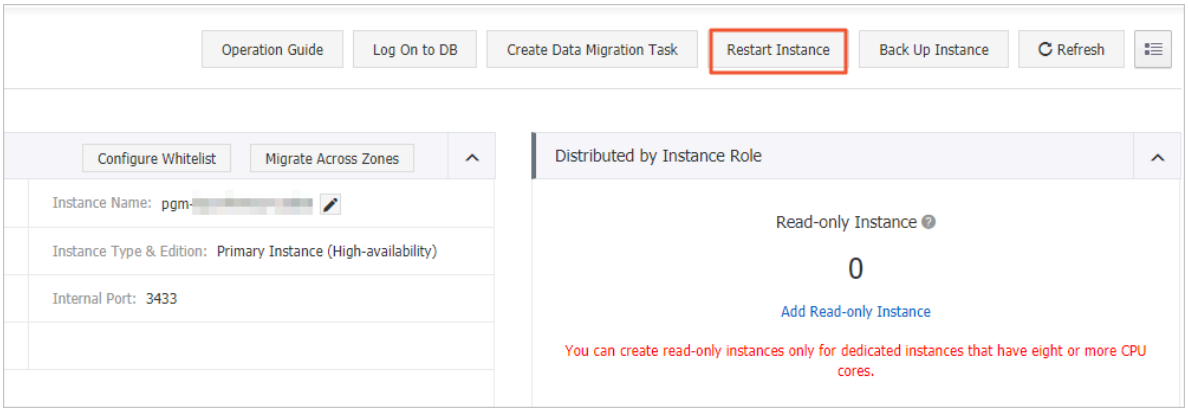
Restarting an RDS instance interrupts connections. Make appropriate arrangements before restarting an RDS instance.

Procedure

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**



3. **Find the target RDS instance. Then, click the instance ID or in the Actions column click Manage.**
4. **In the upper-right corner of the Basic Information page, clickRestart Instance.**



5. In the displayed dialog box, click Confirm.

APIs

API	Description
#unique_88	Used to restart an RDS instance.

6.2 Change the maintenance window of an RDS instance

This topic describes how to change the maintenance window of an RDS instance. To guarantee the stability of ApsaraDB for RDS instances, the back-end system performs maintenance of the instances at irregular intervals. The default maintenance window is from 02:00 to 06:00. You can set the maintenance window to the off-peak period of your business to avoid impacts on business.

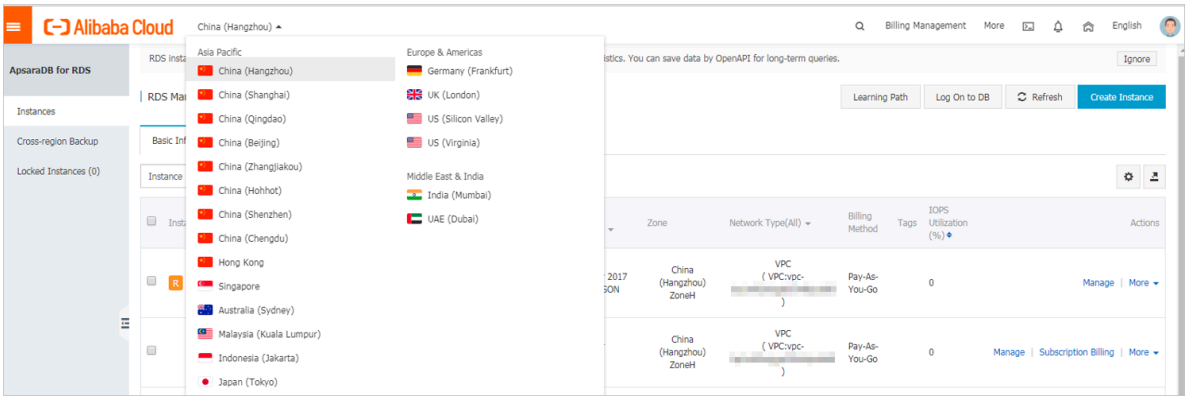
Precautions

- **Before maintenance is performed, ApsaraDB for RDS sends SMS messages and emails to the contacts listed in your Alibaba Cloud accounts.**
- **To guarantee the stability of the maintenance process, the instance enters the Instance Maintaining state before the maintenance time on the day of maintenance. When the instance is in this state, access to data in the database and query operations such as performance monitoring are not affected. However, apart from account and database management and IP address whitelist configuration, modification operations such as upgrade, downgrade, and restart are temporarily unavailable.**
- **During the maintenance window, the instance is disconnected once or twice. Make sure that you configure automatic reconnection policies for your applicatio ns to avoid service disruptions.**

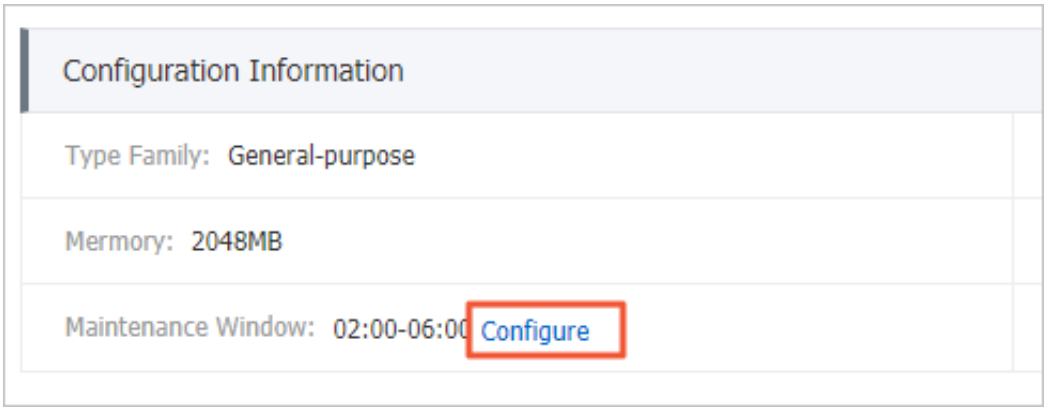
Procedure

1. **Log on to the** *RDS console*.

2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance. Then, click the instance ID, or in the Actions column click Manage.
4. On the Basic Information page, find the Configuration Information section and click Configure to the right of Maintenance Window.



5. Select a maintenance window and click Save.

Note:
The maintenance window is in China Standard Time (UTC +8).

APIs

API	Description
#unique_90	Used to change the maintenance window of an RDS instance.

6.3 Migrate an RDS for MySQL instance across zones in the same region

This topic describes how to migrate an RDS for MySQL instance across zones in the same region.

The attributes, configuration, and connection addresses of the instance remain unchanged after the migration. The time required for the migration varies depending on the data volume of the instance. In typical cases, the migration takes a few hours.

Precautions

During the migration, the connection to your RDS instance remains unavailable for 30 seconds. Make sure that your application can be reconnected to your RDS instance after the migration. The main risks are as follows:

- **The migration changes the virtual IP address. Therefore, we recommend that you use the connection address but not the IP address in your application to establish a connection.**
- **You must delete cached data on the client in a timely manner. Otherwise, data can be read but cannot be written.**
- **A change to the virtual IP address has temporary impact on the availability of DRDS. You must update and view the connection information on the *DRDS console* in a timely manner.**
- **A change to the virtual IP address has temporary impact on *DMS* and *DTS*, which automatically return to normal after the migration is complete.**

Migration scenarios

Migration scenario	Description
Migrate an RDS instance from one zone to another	The zone where the RDS instance is located is overloaded or cannot meet the performance requirements of the instance.

Migration scenario	Description
Migrate an RDS instance from one zone to multiple zones	The master and slave nodes are located in different equipment rooms in different zones to enhance disaster tolerance. A multi-zone instance is superior to a single-zone instance because it can survive more disasters. For example, a single-zone instance can survive server and rack faults while a multi-zone instance can survive equipment room faults.
Migrate an RDS instance from multiple zones to one zone	This scenario is provided to meet the requirements of specific functions.

Fees

This function is free of charge. No fee is charged even when you migrate an RDS instance from one zone to multiple zones.

Prerequisites

Instance types: MySQL 5.5, MySQL 5.6, and MySQL 5.7 (with local SDDs).

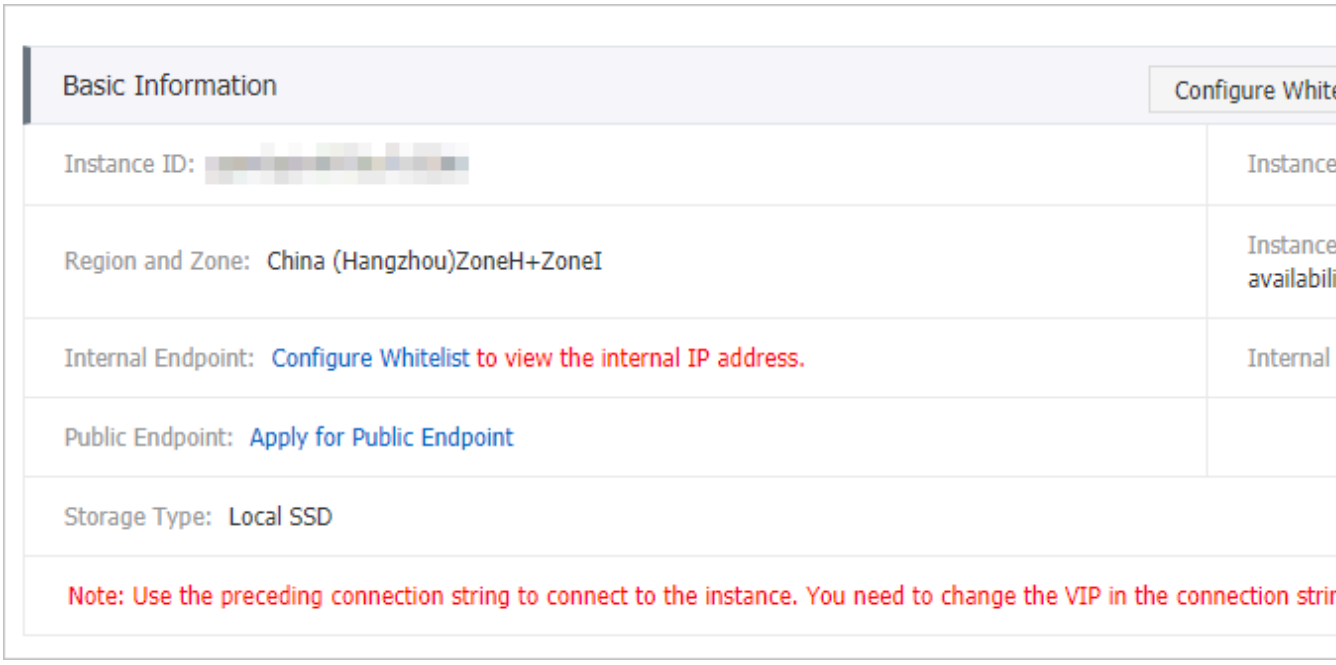
Regions: This function is available only when the region where RDS instance is located consists of multiple zones. For more information about regions and zones, see *Regions and zones*.

Procedure

1. Log on to the *RDS console*.
2. In the upper-left corner, select the region where the target RDS instance is located.

3. Find the target RDS instance and click its ID.

4. **In the Basic Information section of the Basic Information page, click Migrate Across Zones.**

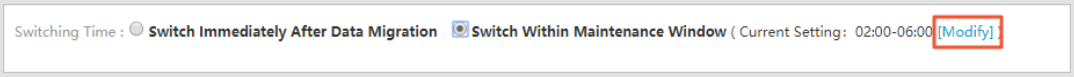


5. **In the displayed dialog box, specify the destination zone, VSwitch, and migration time, and click OK.**

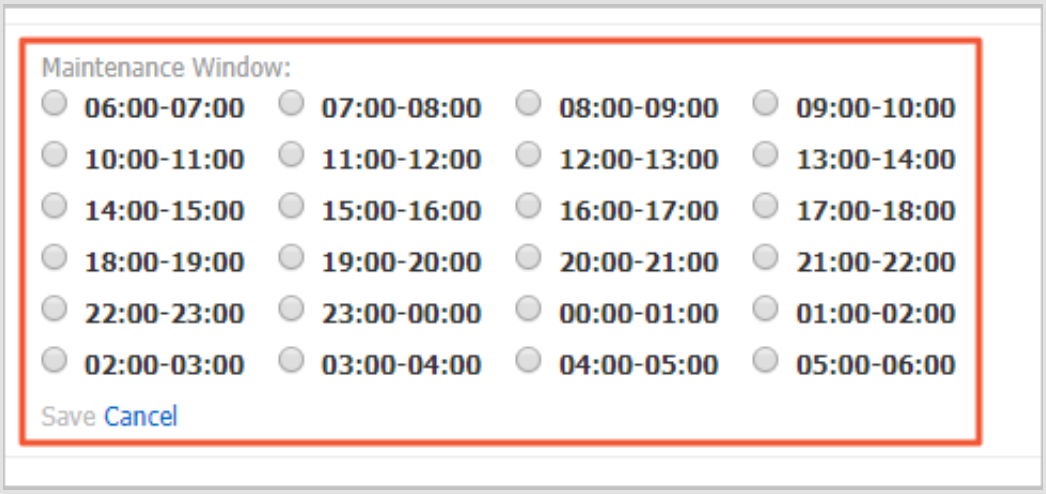
Note:

If you want to change the maintenance window, follow these steps:

a. **Click Change.**



b. **In the Configuration Information section, specify the maintenance window and click Save.**



c. Refresh the page, and perform the migration again.

APIs

API	Description
#unique_92	Used to migrate an RDS instance across zones.

6.4 Manually or automatically switch over services between the RDS for MySQL master and slave instances

This topic describes how to manually or automatically switch over services between the RDS for MySQL master and slave instances. After the switchover, the master instance becomes the slave instance.

Prerequisites

The master instance is in the High-availability or Enterprise Edition.

Note:

- RDS instances in the Basic Edition do not have slave instances, and therefore do not support service switchovers.
- Only RDS instances in the High-availability Edition allow you to temporarily disable the automatic switchover function.

Context

- Automatic switchover: the default switchover mode. When the master instance becomes faulty, your RDS services are automatically switched over to the slave instance.
- Manual switchover: You can manually switch over services between the master and slave instances even when the automatic switchover function is enabled.

Note:

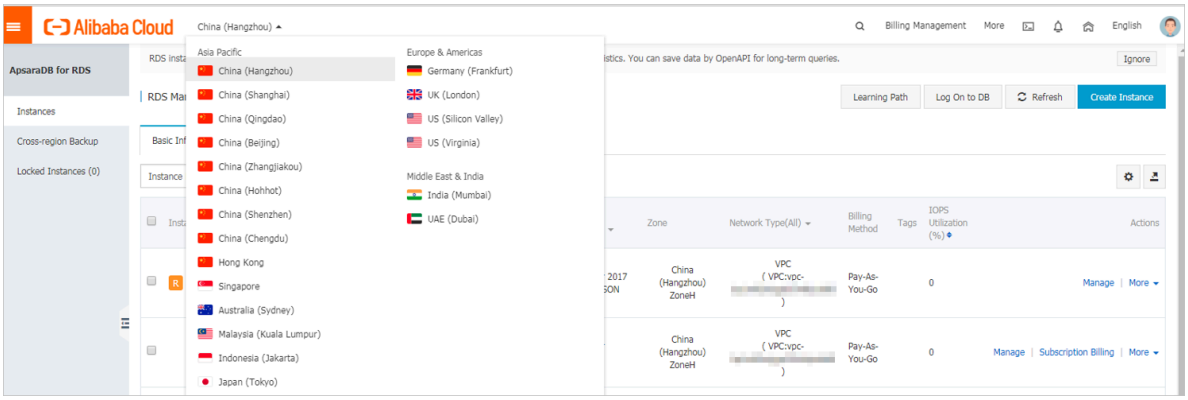
In the High-availability edition, an RDS instance (referred to as the master instance) has a slave instance. Data is synchronized in real time between the master and slave instances. You can only access the master instance. The slave instance is used only as a backup and does not provide services. After a service switchover, the master instance is degraded to the slave instance.

Precautions

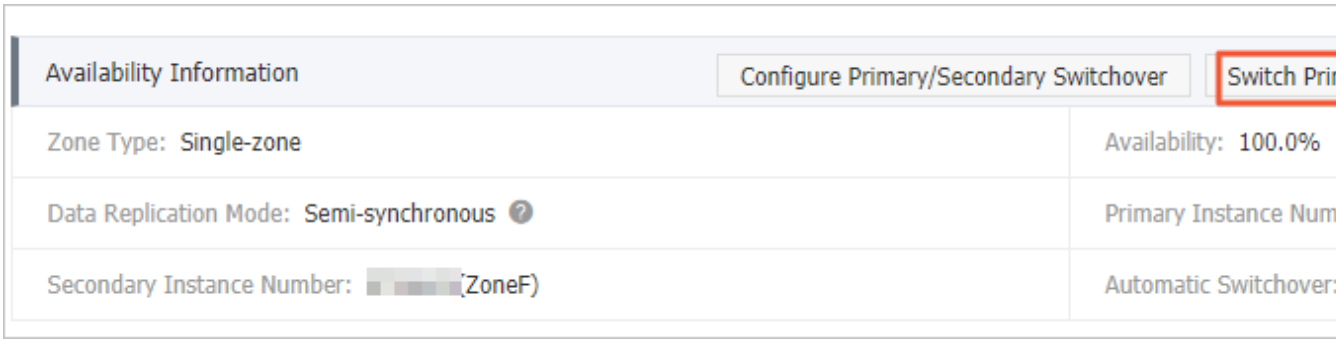
- **Services may be disconnected during a switchover. Make sure that you configure automatic reconnection policies for your applications to avoid loss of services.**
- **If read-only RDS instances are mounted to your RDS instance, the data in the read-only RDS instances shows a few minutes' delay after a switchover. This is because it takes time to reestablish replication links and synchronize incrementa l data.**

Manually switch over services between the master and slave instances

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**



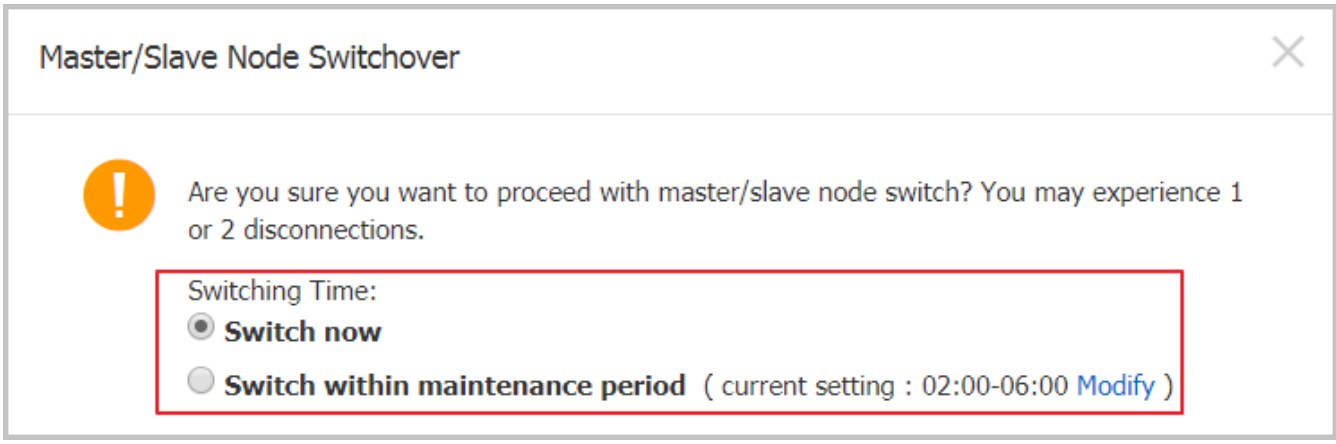
3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Service Availability.**
5. **In the Availability Information section, click Switch Primary/Secondary Instance.**



6. **Select the time at which you want to perform a switchover, and click OK.**

Note:

During the switchover, you cannot perform operations such as managing databases and accounts and changing the network type. Therefore, we recommend that you select Switch Within Maintenance Window.

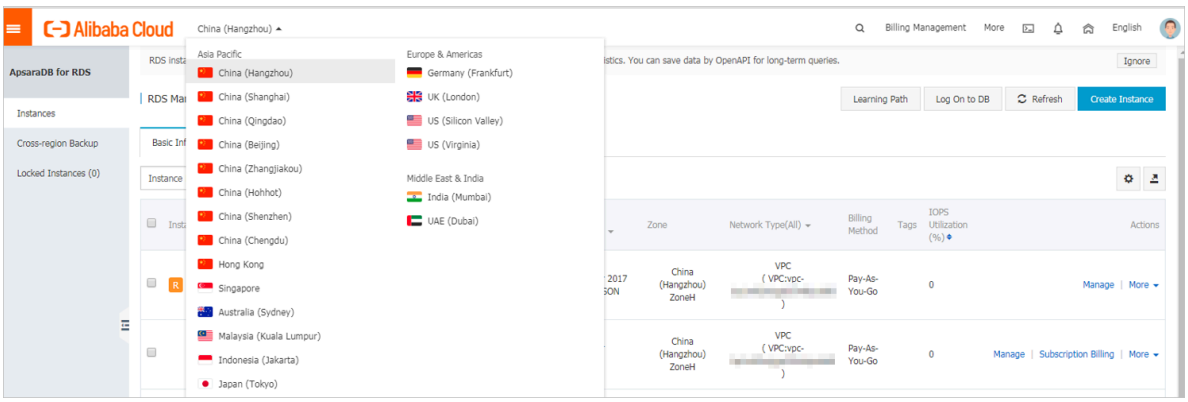


Temporarily disable automatic switchover

The automatic switchover function is enabled by default. In such case, services can be automatically switched over to the slave instance if the master instance becomes faulty. You can disable the automatic switchover function in one of the following situations:

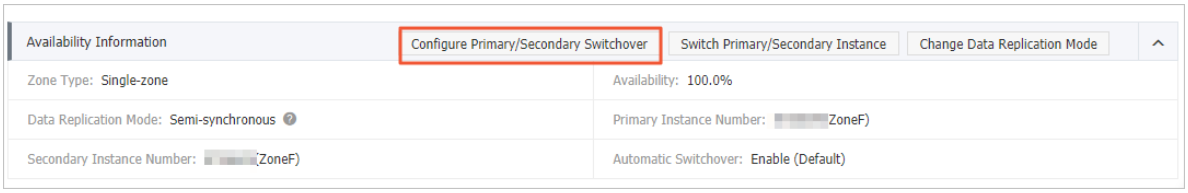
- **A sales promotion, during which you do not want a switchover to affect service availability.**
- **An important application upgrade, during which you do not want a switchover to incur any unexpected issues.**
- **A major event that requires stable system operation, for example, a national holiday, during which you do not want a switchover to affect system stability.**

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**



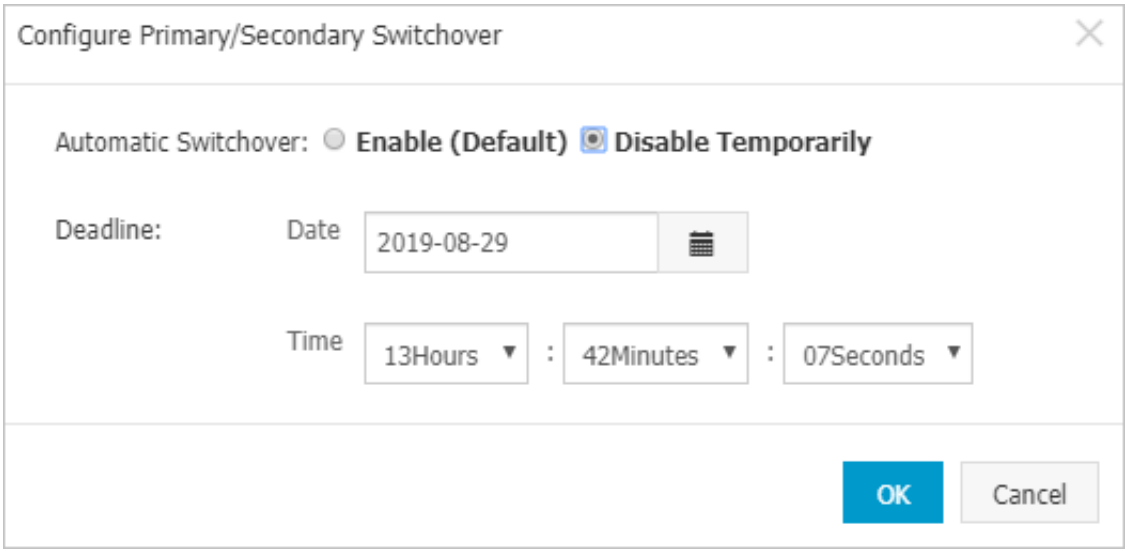
3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Service Availability.
5. In the Availability Information section, click Configure Primary/Secondary Switchover.

> **Note:**
> If the Configure Primary/Secondary Switchover button is unavailable, make sure that the RDS instance is in the High-availability Edition.

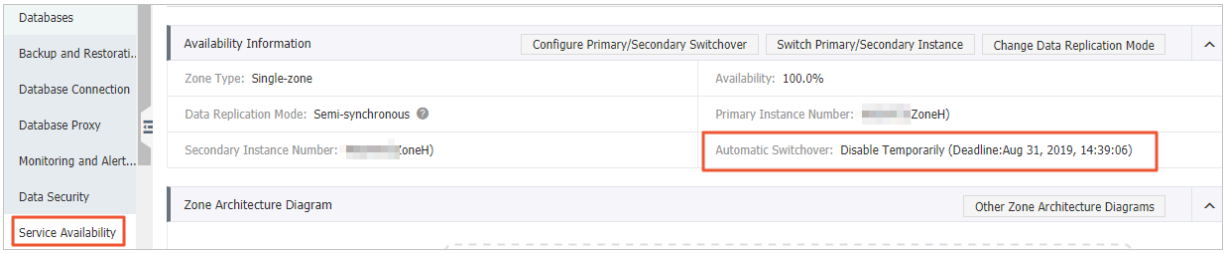


6. Select Disable Temporarily, set Deadline, and click OK.

> **Note:**
> - When the specified Deadline arrives, the automatic switchover function is restored to enabled for the RDS instance.
> - By default, the automatic switchover function is disabled for one day. You can set the Deadline parameter to 23:59:59 seven days later at tops.



After the setting is complete, you can go to the Service Availability page to check the deadline for disabling the automatic switchover function.



6.5 Change the network type of an RDS for MySQL instance

This topic describes how to change the network type of an RDS for MySQL instance.

Network types

- **Classic network: Instances in a classic network are not isolated. Access control is implemented for instances by using whitelists.**
- **Virtual Private Cloud (VPC): A VPC is an isolated network environment. We recommend that you use VPC because it is more secure.**

  You can customize the routing table, IP address range, and gateway of the VPC. To smoothly migrate applications to the cloud, you can use a leased line or VPN to connect your own data center to a VPC on the cloud to make a virtual data center.

Note:

- **You can use the classic network or VPC and switch between the network types for free.**
- **For PostgreSQL instances, you must switch the IP whitelist mode to the enhanced whitelist mode before switching the network type. For more information, see** *Switch to the enhanced whitelist mode for an RDS for MySQL instance*.

Switch from VPC to classic network

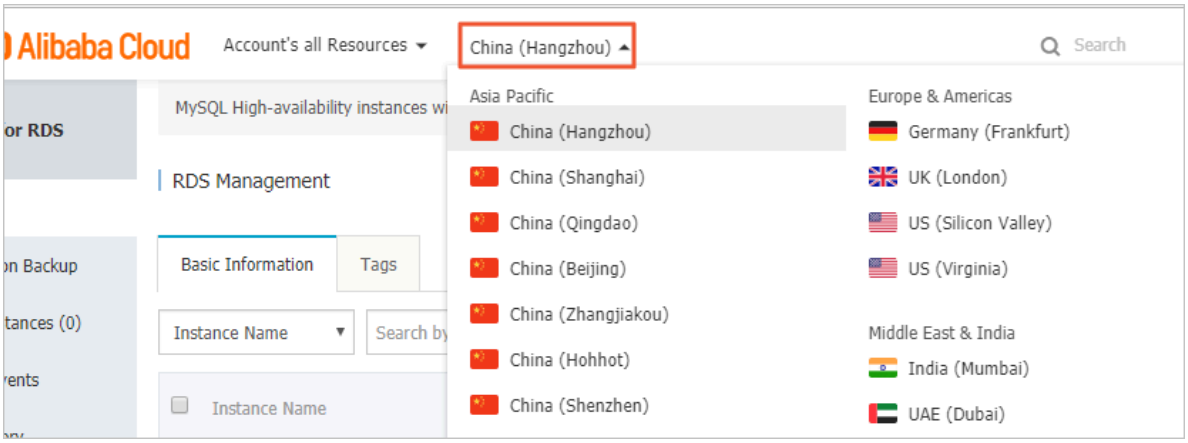
Precautions

- **After the network type of an RDS instance is switched to classic network, the endpoints remain unchanged, but the corresponding IP addresses change.**
- **After the network type of an RDS instance is switched to classic network, ECS instances in VPCs cannot access the RDS instance by using the internal endpoint . Make sure that you change the endpoint on the application.**

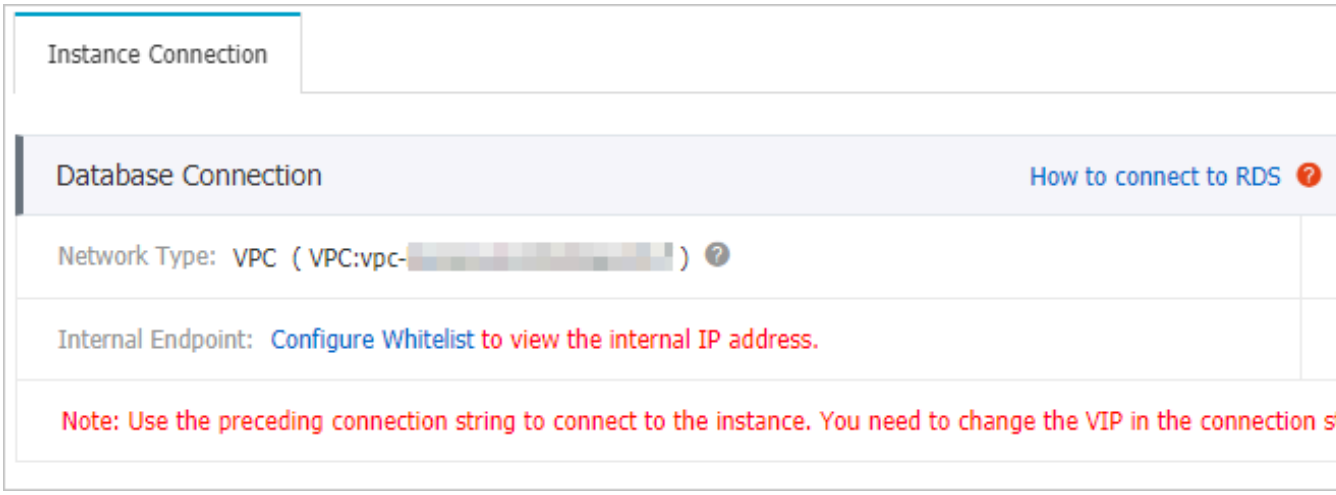
- Switching the network type may result in a disconnection of 30 seconds. To avoid impacts that arise from this operation, we recommend that you perform the switching during off-peak hours, or configure automatic reconnection policies for your application.
- Instances of PostgreSQL 11 High-availability Edition (cloud disk), PostgreSQL 10 High-availability Edition (cloud disk), and PostgreSQL 10 Basic Edition do not support the classic network. Therefore, you cannot switch these instances to the classic network.

Procedure

1. Log on to the *ApsaraDB for RDS console*.
2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Database Connection.
5. In the Database Connection section, click Switch to Classic Network.



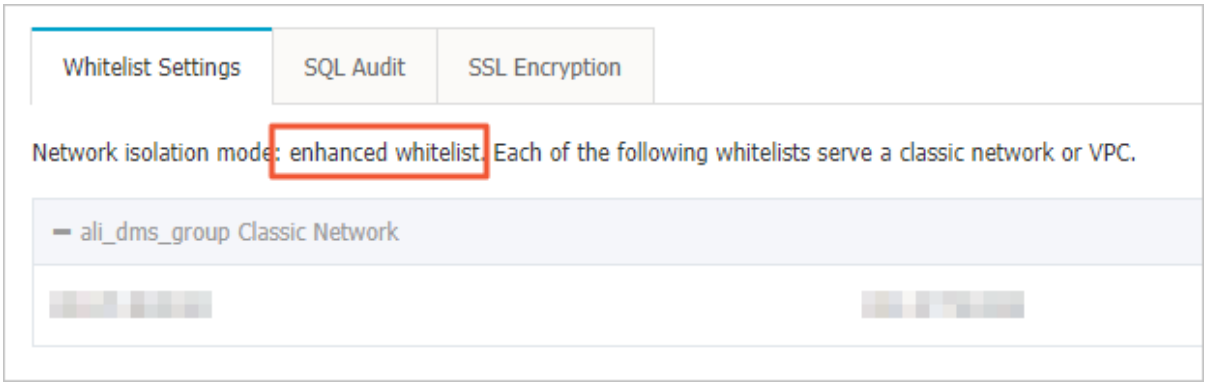
6. **In the message that appears, click OK.**

   After the network type is switched, only ECS instances in classic networks can access the RDS instance over the internal network. Make sure that you configure the endpoint of the RDS instance on the ECS instance in the classic network.

7. **Configure the whitelist of the RDS instance to allow access from the ECS instance over the internal network.**

   - **If the RDS instance applies the standard whitelist mode, as shown in the following figure, you must add the internal endpoint of the ECS instance in the classic network to any whitelist of the RDS instance.**

   - **If the RDS instance applies the *enhanced whitelist mode*, as shown in the following figure, you must add the internal endpoint of the ECS instance in the classic network to the default classic network whitelist of the RDS instance. If there is no classic network whitelist, you must create a new whitelist.**

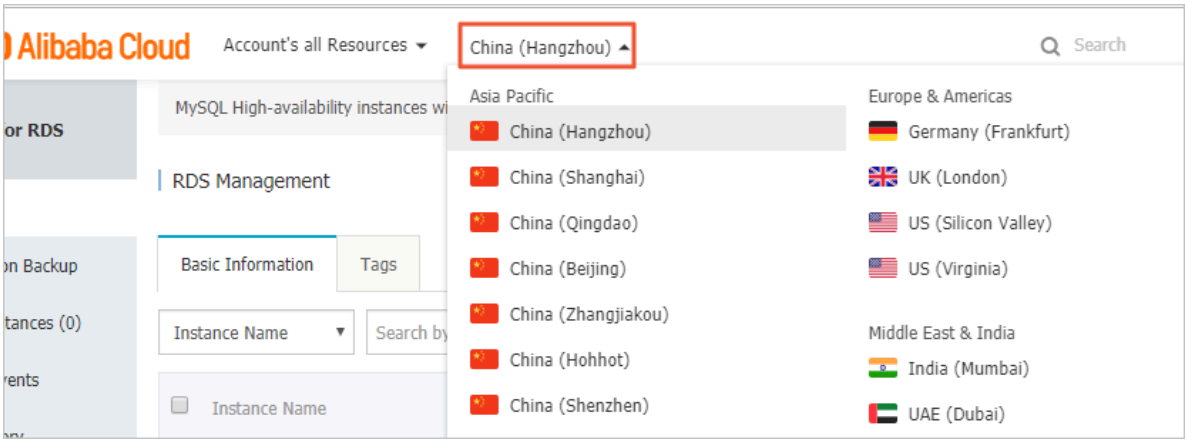
     

Switch from classic network to VPC

Procedure

1. **Log on to the *ApsaraDB for RDS console*.**

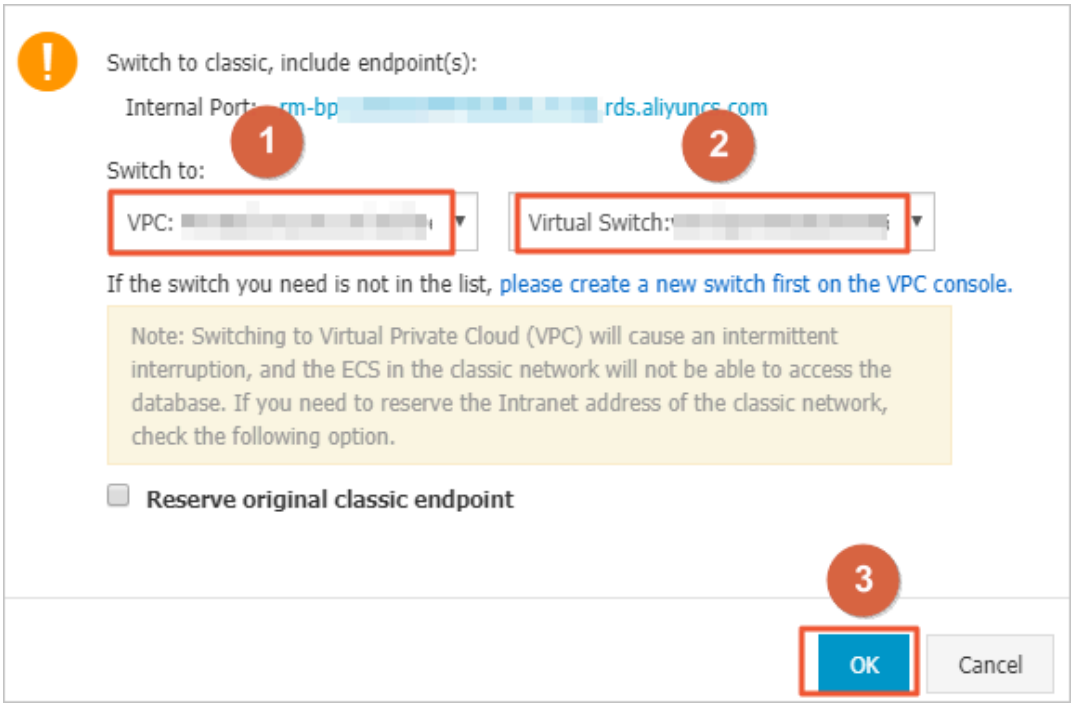
2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Database Connection .
5. Click Switch to VPC.
6. In the dialog box that appears, select a VPC and a VSwitch, and specify whether to retain the classic network address.
   - Select a VPC. We recommend that you select the VPC where your ECS instance is located. Otherwise, the ECS and RDS instances cannot connect to each other

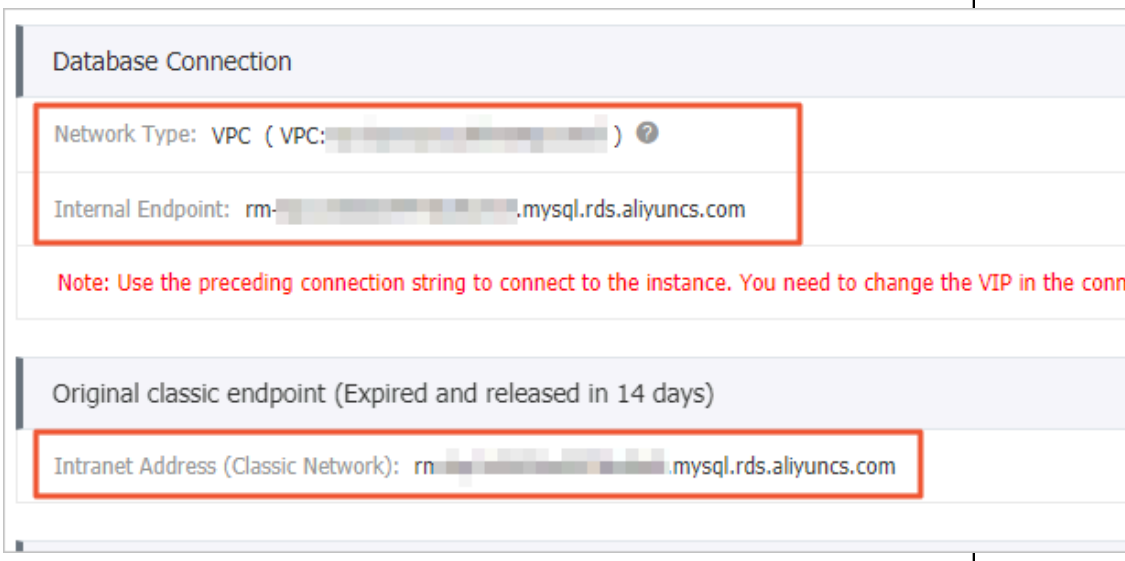
over the internal network unless *Express Connect* or *VPN Gateway* are created to connect the two VPCs.

- Select a VSwitch. If there is no VSwitch in the VPC that you select, as shown in the following figure, you must create a VSwitch in the zone where the instance is located. For more information, see *Manage VSwitches*.



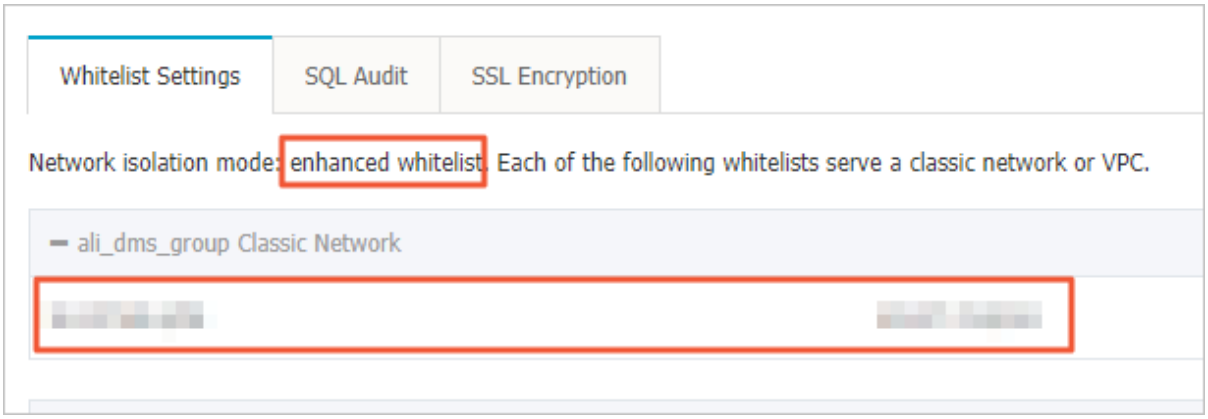
- Select or clear Reserve Original Classic Endpoint as needed. The following table describes the details.

Action	Description
Clear	The classic network address is not retained. The original classic network address is changed to the VPC address. If you do not retain the classic network address, the RDS instance will be disconnected for 30 seconds, and the access from the ECS instance in the classic network to the RDS instance over the internal network is immediately disconnected when you switch the network type.

Action	Description
Select	The classic network address is retained, and a new VPC address is generated, as shown in the following figure. It indicates that the *hybrid access mode* is enabled, and the RDS instance can be accessed by ECS instances in both a classic network and a VPC. If you retain the classic network address, the RDS instance will not be disconnected when you switch the network type. The internal access from the ECS instance in the classic network to the RDS instance is only disconnected when the classic network address expires. Before the classic network address expires, make sure that the VPC address has been configured in the ECS instance in the VPC to smoothly migrate your services to the VPC. The system will send an SMS message to the phone number bound to your Alibaba Cloud account every day in the seven days before the classic network address expires.  For more information, see *Configure a hybrid access solution to smoothly migrate an RDS instance from the classic network to a VPC*.

7. Add the internal IP address of the ECS instance in the VPC to the VPC whitelist of the RDS instance, so that the ECS instance can access the RDS instance over the

internal network, as shown in the following figure. If there is no VPC whitelist, you must create a new whitelist.



8. Perform one of the following operations as needed:
   - If you retain the classic network address, you must configure the VPC address of the RDS instance in the ECS instance that is in the VPC.
   - If you do not retain the classic network address, the access from the ECS instance in the classic network to the RDS instance over the internal network is immediately disconnected when you switch the network type. You must configure the VPC address of the RDS instance in the ECS instance that is in the VPC.

> **Note:**
> If you need to use the ECS instance in the classic network to access the RDS instance in the VPC, you can use the *ClassicLink* function or migrate the ECS instance to the VPC.

Related operations

Operation	Description
#unique_99	Used to switch the network type of an RDS instance.

6.6 Change the data replication mode of an RDS for MySQL instance

This topic describes how to change the data replication mode of an RDS for MySQL instance and its slave instance to improve service availability

This function is available to the High-availability Edition of MySQL 5.5, 5.6, and 5.7.

Note:
In the MySQL 5.6 or 5.7 Enterprise Edition, you cannot change the replication mode.

Overview of data replication

- **Semi-synchronous**

  The updates initiated from your application are performed on the master RDS instance. Then the corresponding logs are synchronized to the slave RDS instance. The system regards that the transactions are submitted before the updates recorded in the logs are performed on the slave RDS node.

  If the slave RDS instance is unavailable or communication between the master and slave RDS instances is abnormal, the semi-synchronous data replication mode is degraded to the asynchronous data replication mode.
- **Asynchronous**

  When your application initiate an update request for adding, deleting, or modifying data, the master RDS instance responds to your application immediately and replicates data to the slave RDS instance asynchronously after completing the requested operation. Therefore, in asynchronous data replicatio n mode, the operations on the master RDS instance are not affected even when the slave RDS instance is unavailable. Additionally, the probability that data inconsistencies are incurred by unavailability of the master RDS instance is low.

Prerequisites

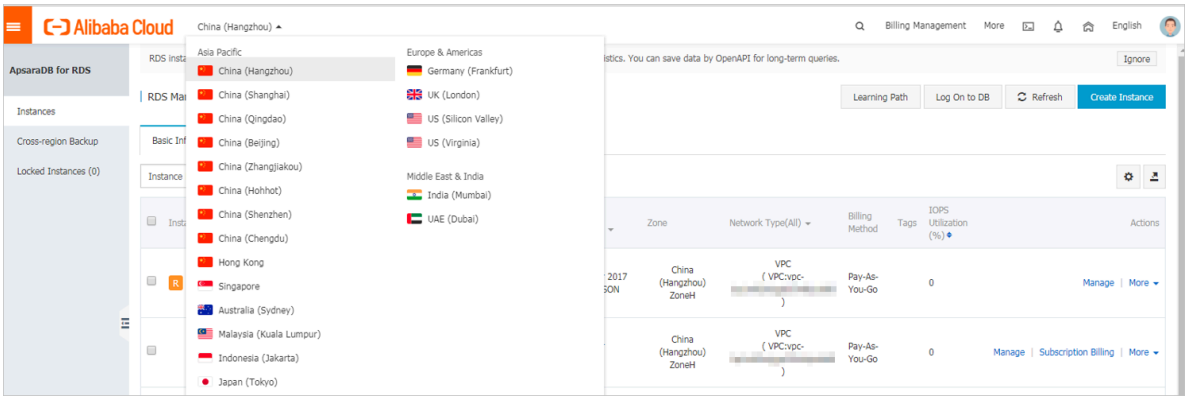
The DB engine version and edition are one of the following:

- **MySQL 8.0 High-availability Edition (with local SSDs)**
- **MySQL 5.7 High-availability Edition (with local SSDs)**
- **MySQL 5.6 High-availability Edition**
- **MySQL 5.5**

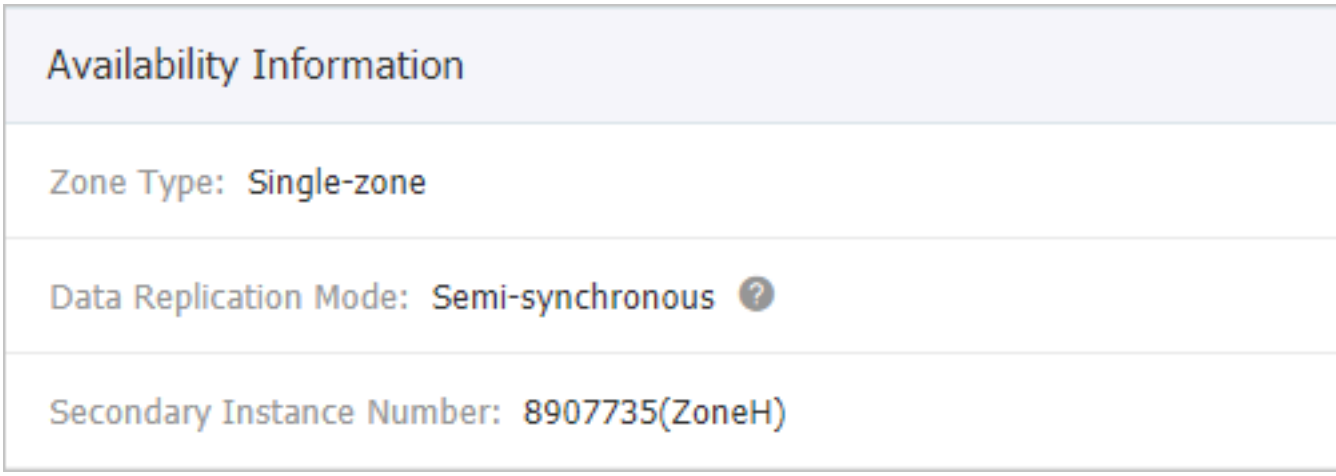
Procedure

1. **Log on to the** *RDS console*.

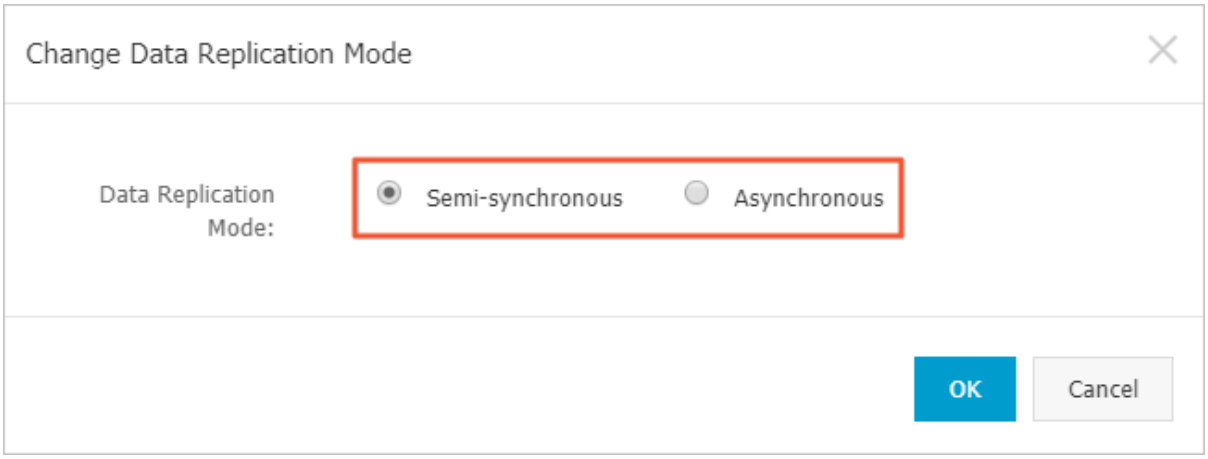
2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Service Availability.
5. In the Availability Information section, click Change Data Replication Mode.



6. In the displayed dialog box, select a data replication mode and click OK.



APIs

API	Description
#unique_101	Used to change the HA mode and data replication mode of an RDS instance.

6.7 Release an RDS for MySQL instance

This topic describes how to release an RDS for MySQL instance, which can use the pay-as-you-go or subscription billing method.

> **Note:**
> **After an RDS instance is released, its data is deleted immediately. We recommend that you back up the instance data before you release the instance.**

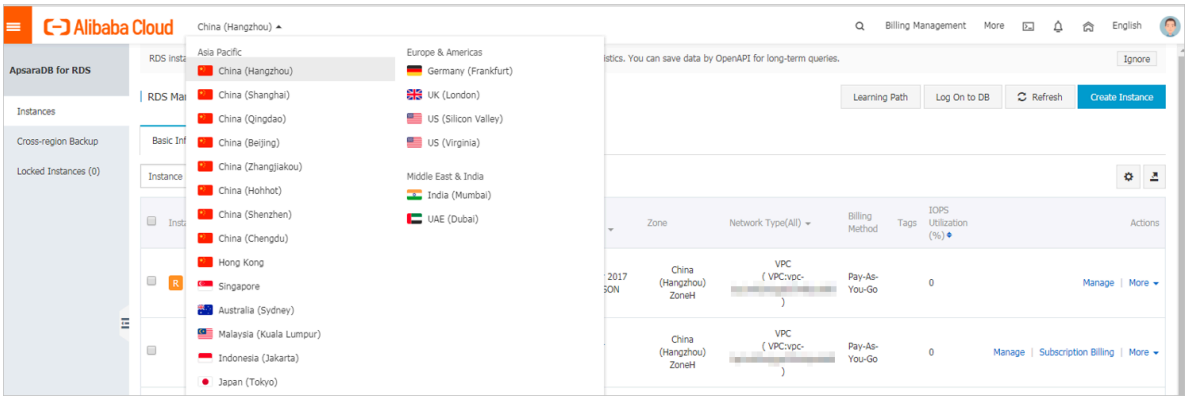
Release a pay-as-you-go RDS instance

Precautions

If the RDS instance you want to release is the last read-only instance for a master RDS instance, you must first disable the read/write splitting function. For more information, see *Disable read/write splitting for an RDS for MySQL instance*.

Procedure

1. **Log on to the *RDS console*.**
2. **Select the target region.**



3. Use one of the following two methods to open the Release Instance dialog box:

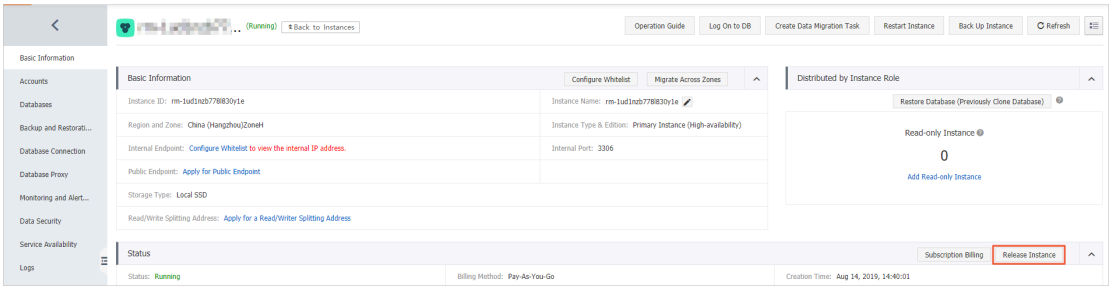
   - Method 1:

     Find the target RDS instance and in the Actions column choose More > Release Instance.

   - Method 2:

     a. Find the target RDS instance and click the instance ID.

     b. On the Basic Information page, find the Status section and click Release Instance.



4. In the Release Instance dialog box, click Confirm.

Release a subscription RDS instance

You can *open a ticket* to apply for releasing a subscription RDS instance.

FAQ

Will my RDS services be affected if I release a read-only instance?

Yes, we recommend that you set the *read weight* of the read-only instance to 0 before you release a read-only instance.

> **Note:**
> After you release a read-only instance, the cached connection still points to this read-only instance. To enable the system to distribute read requests to the other read-only instances, you must reestablish a connection.

APIs

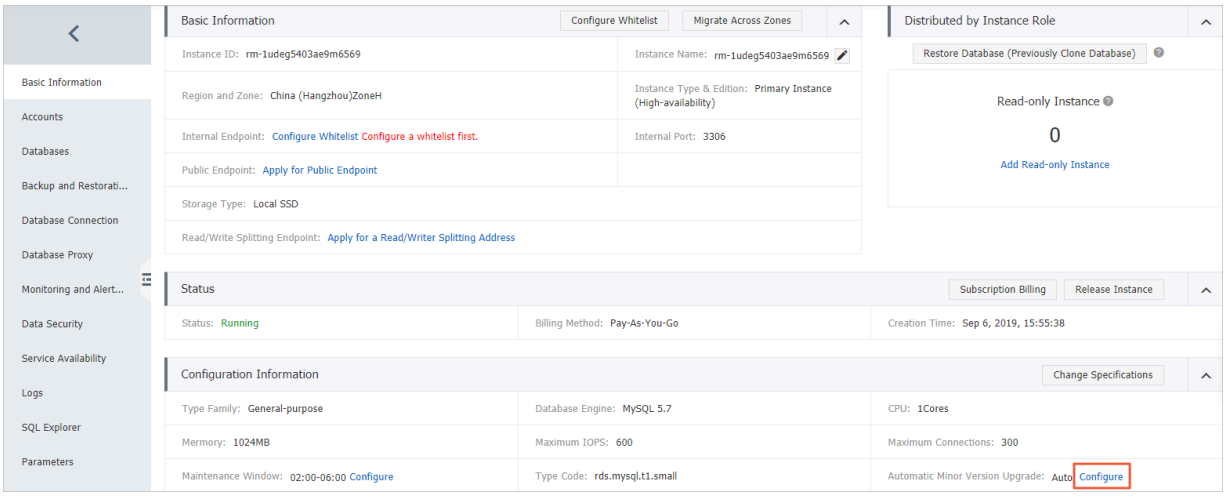
API	Description
#unique_105	Used to release a pay-as-you-go RDS instance. (A subscription RDS instance cannot be released by calling an API action.)

6.8 Upgrade the kernel minor version of an RDS for MySQL instance

This topic describes how to upgrade the kernel minor version of an RDS for MySQL instance. The upgrade of the kernel minor version helps to improve system performance, introduce new functions, and fix defects.

Overview

The automatic minor version upgrade function is enabled by default. You can navigate to the Basic Information page on the RDS console and then in the Configuration Information section view information about Automatic Minor Version Upgrade.



- **Auto: In this mode, Alibaba Cloud sends you notifications when a new kernel minor version is released. The automatic upgrade is performed during the specified *maintenance window*.**
- **Manual: In this mode, Alibaba Cloud sends you notifications when the lifecycle of the kernel minor version used by your RDS for MySQL instance ends. You are**

notified that you must upgrade your RDS for MySQL instance to the latest kernel minor version within one month.

Note:
When the lifecycle of a kernel minor version ends, this kernel minor version is deprecated.

Precautions

- The kernel minor version upgrade incurs an restart of your RDS instance. During the restart, your RDS instance may be unavailable for 30 seconds or less. We recommend that you perform the upgrade during off-peak hours or make sure that your application can automatically reconnect to your RDS instance.
- The kernel minor version of your RDS instance cannot be degraded after being upgraded.
- After you upgrade the configuration of your RDS instance, the kernel minor version of your RDS instance is upgraded accordingly.

Prerequisites

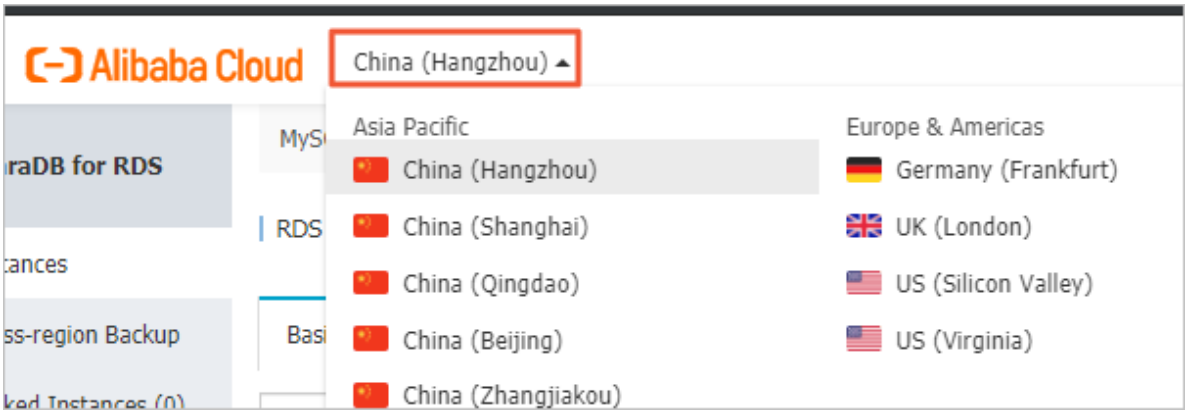
The used DB engine version and edition are one of the following:

- MySQL 8.0 High-availability Edition (with local SSDs)
- MySQL 8.0 Basic Edition
- MySQL 5.7 High-availability Edition (with local SSDs)
- MySQL 5.7 Basic Edition
- MySQL 5.6
- MySQL 5.5

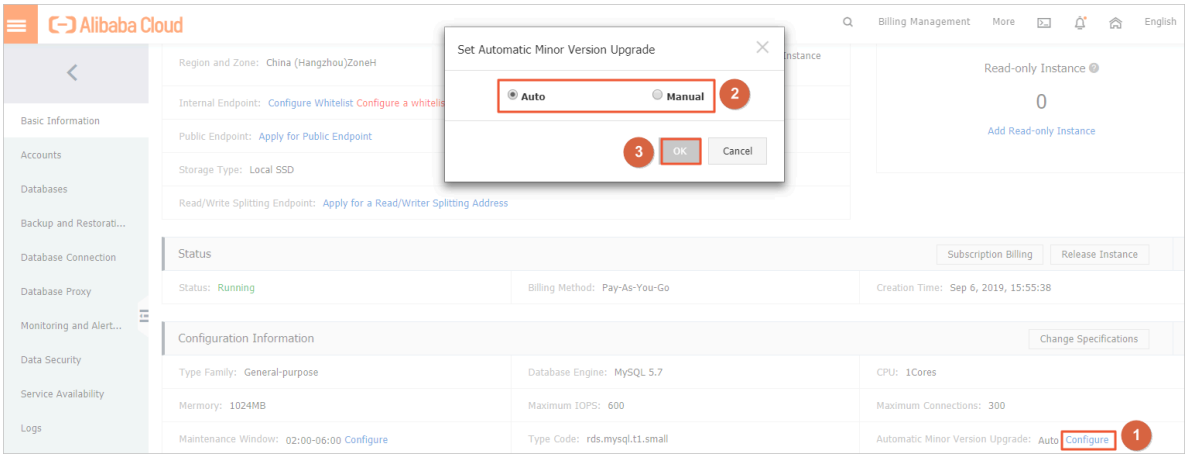
Set automatic minor version upgrade

1. Log on to the *RDS console*.

2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. On the Basic Information page, find the Configuration Information section, and click Configure to the right of Automatic Minor Version Upgrade.
5. In the displayed dialog box, select Auto or Manual and click OK.



6.9 Change the configuration of an RDS for MySQL instance

This topic describes how to change the configuration of an RDS for MySQL instance, including changing the edition, specifications, storage capacity, storage class, and zone.

You can upgrade or downgrade the configuration of an RDS for MySQL instance at any time regardless of whether the instance uses the subscription or pay-as-you-go billing method. The new configuration takes effect immediately after you complete the configuration upgrade or downgrade.

Configuration items

If you want to horizontally scale the read capability of an RDS for MySQL instance, you can create read-only instances. For more information, see *Read-only instances overview* **and** *Create an RDS for MySQL read-only instance*.

Configuration item	Description
Edition	· MySQL 5.6: You can change the DB engine edition from High-availability Edition to Enterprise Edition or from Enterprise Edition to High-availability Edition. **Note:** If you use the MySQL 5.6 DB engine, the network type of the RDS instance must be Classic. · MySQL 5.7: You can change the DB engine edition from Basic Edition to High-availability Edition.
Specifications (type)	You can change the specifications of any RDS instance.
Storage capacity	You can increase the storage capacity of any RDS instance. You can only decrease the storage capacity of a subscription-based instance with local SSDs during *instance renewal*. **Note:** · For information about the capacity range, see *#unique_24*. · You cannot decrease the storage capacity if the RDS instance uses cloud SSDs. · If the storage capacity range of the current specifications cannot meet your requirements, you can change the specifications.
Storage class	When you change the MySQL 5.7 DB engine of an RDS instance from Basic Edition to High-availability Edition, the storage class changes from SSD to local SSD.

Configuration item	Description
Zone	When you switch the MySQL 5.6 DB engine of an RDS instance between High-availability Edition to Enterprise Edition, you must also change the zone of the RDS instance. Note: Changing the zone requires data migration. A larger data volume indicates a longer migration time.

Note:
Changing the preceding configuration does not change the endpoints of the RDS instance.

Billing

For more information, see *#unique_108*.

Prerequisites

Your Alibaba Cloud account does not have an unpaid renewal order.

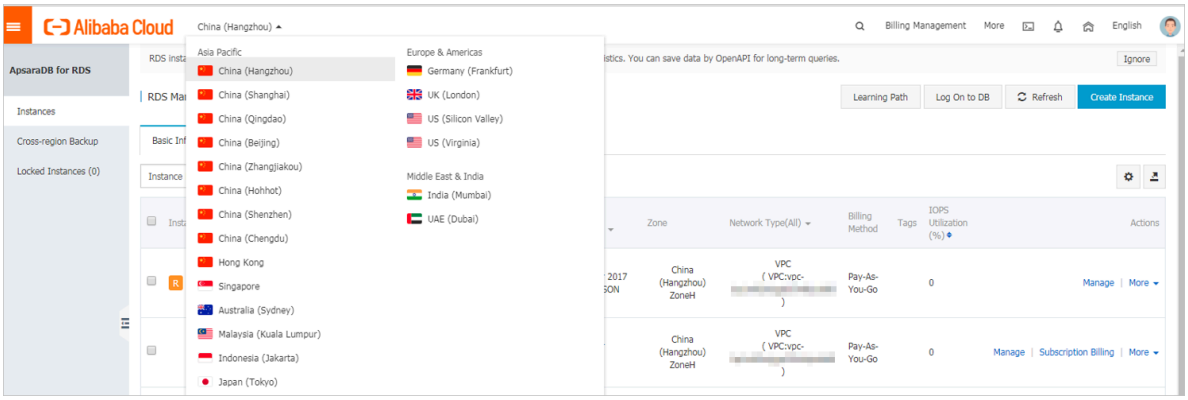
Precautions

- When the new configuration is taking effect, the RDS instance may be disconnected for about 30 seconds and most operations related to databases, accounts, and networks cannot be performed. Therefore, we recommend that you change the configuration during off-peak hours or make sure that your application can automatically reconnect to the RDS instance.
- If the RDS instance is in the Basic Edition (which has no slave node as hot backup), it becomes unavailable for a long time during the configuration change. Therefore, if you have high requirements for database availability, we recommend that you select an edition higher than the Basic Edition. For example, you can select the High-availability Edition.

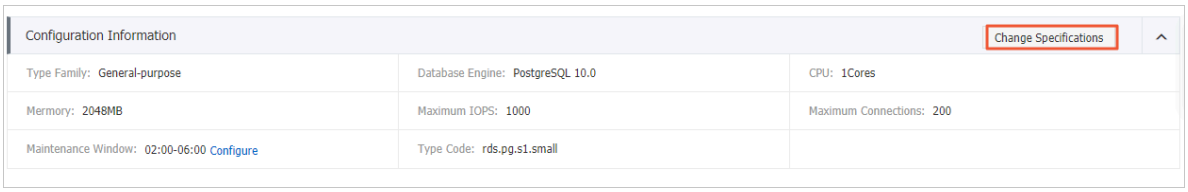
Procedure

1. Log on to the *RDS console*.

2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. On the Basic information page, find the Configuration Information section and click Change Specifications.

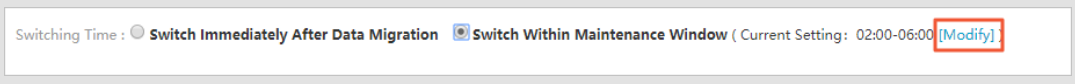


5. Optional. If the RDS instance uses the subscription billing method, click Next in the displayed dialog box.
6. On the Change Specifications page, change the instance configuration. For more information, see *Configuration items*.
7. Specify the time at which you want to change the configuration.
   - Switch Immediately After Data Migration: Change the configuration immediately after the data migration.
   - Switch Within Maintenance Window: Change the configuration during the *maintenance window*.

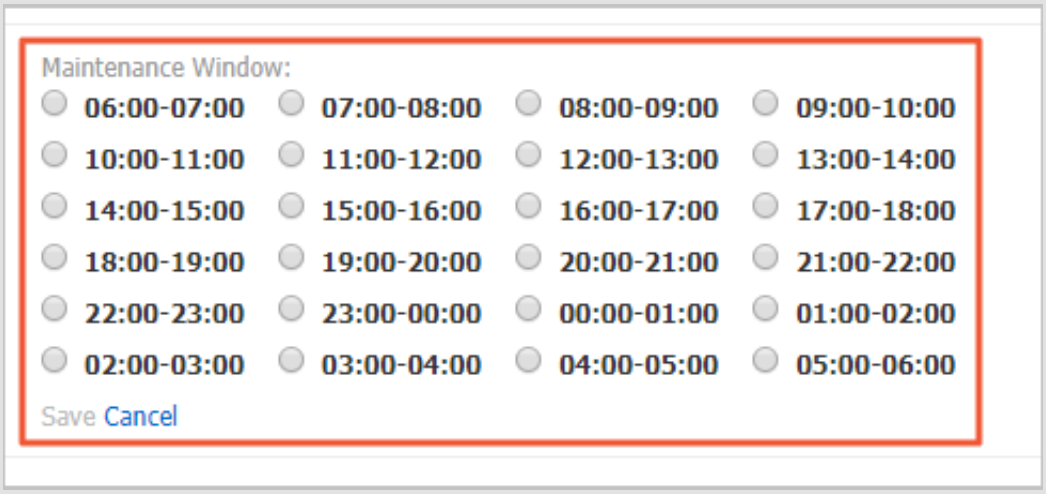
Note:

To change the maintenance window, follow these steps:

a. Click Modify.



b. In the Configuration Information section, select a maintenance window and click Save.



c. Go back to the Change Specifications page, refresh the page, and change the configuration again.

8. Select Terms of Service, Service Level Agreement, and Terms of Use and click Confirm.

FAQ

1. How can I change the storage class (local SSD, SSD, or ESSD) of an RDS instance?

   See *#unique_109*

2. Do I need to migrate data if I only want to expand the storage capacity of an RDS instance?

   Check whether the server where the RDS instance is located provides sufficient storage capacity for expansion. If yes, you do not need to migrate data and can directly expand the storage capacity. If no, you must migrate data to a server that provides sufficient storage capacity before you expand the storage capacity.

6.10 Reconfigure parameters for an RDS for MySQL instance

This topic describes how to use the console or API to view and reconfigure some parameters for an RDS for MySQL instance. You can also use the console to query the parameter reconfiguration history.

Precautions

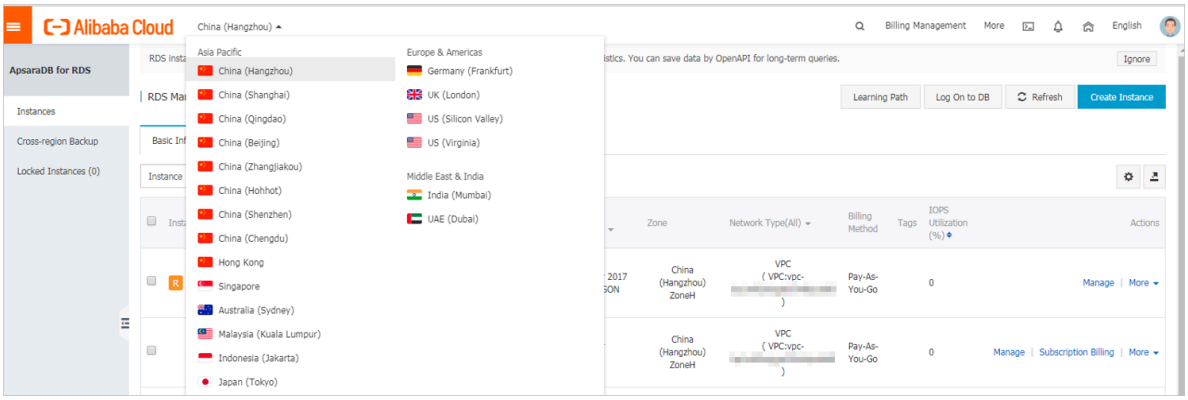
- When you reconfigure parameters on the Parameters page, see the Value Range column corresponding to each parameter.

- **After you reconfigure certain parameters, you must restart the RDS instance for the changes to take effect. For more information, see the Restart column on the Parameters page. A restart disconnects the RDS instance. We recommend that you make appropriate service arrangements before you restart an RDS instance. Proceed with caution.**

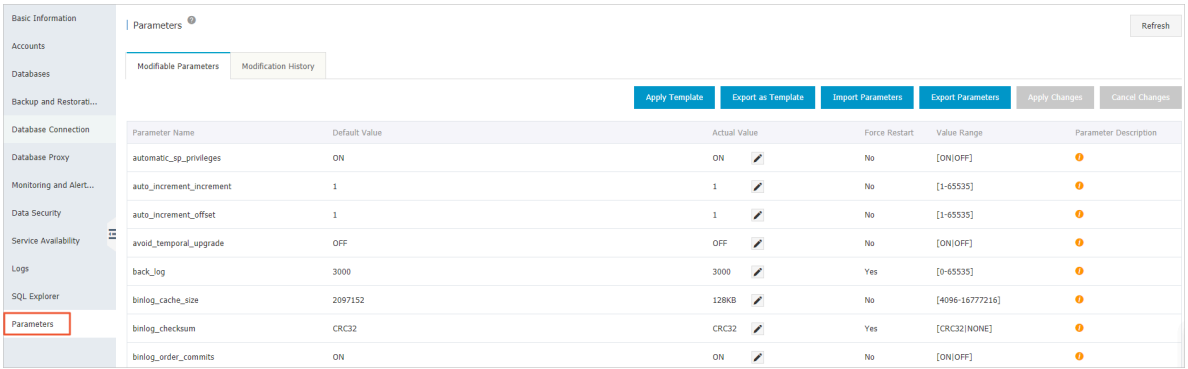
Reconfigure parameters

If you want to reconfigure the parameters of an RDS instance in batches, we recommend that you *use a parameter template*.

1. **Log on to the *RDS console*.**
2. **Select the target region.**



3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Parameters.**



5. **On the Modifiable Parameters tab, reconfigure one or more parameters as needed.**

   To apply a parameter template to the RDS instance, follow these steps:

Note:

If applying the parameter template requires a restart, we recommend that you perform the operation during off-peak hours and make sure that your application can automatically reconnect to the RDS instance.

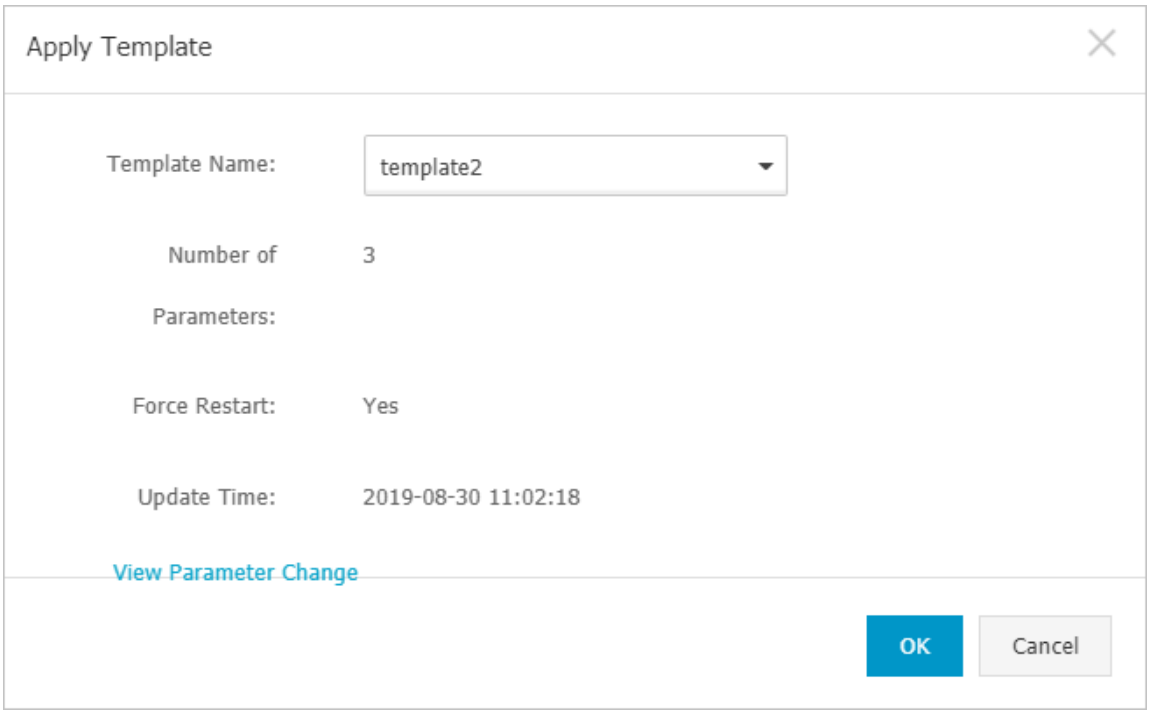
a) **Click Apply Template.**

b) **In the Apply Template dialog box, select the parameter template you want to apply, and click OK.**

Note:

- **You can view the number of parameters in the template and whether the template requires a restart. For more information about the parameter changes, click View Parameter Change.**
- **If you cannot find the template, check whether the template is in the same region as the RDS instance. If they are in different regions, you can**

replicate the template to the region where the RDS instance is located. For more information, see *Operate parameter templates*.



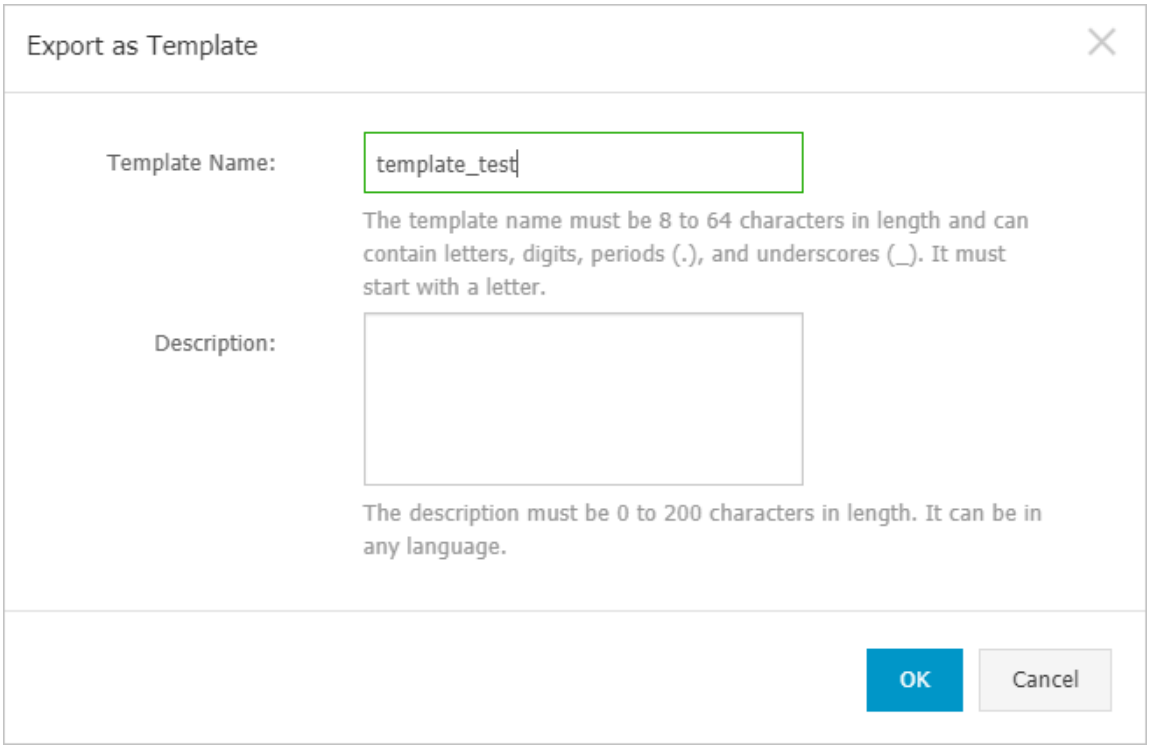
To export the parameter settings of the RDS instance as a parameter template, follow these steps:

a) Click Export as Template.

b) Set the following parameters.

Parameter	Description
Template Name	The name of the parameter template. The template name must start with a letter and can contain 8 to 64 characters including letters, digits, decimal points (.), and underscores (_).

Parameter	Description
Description	The description of the parameter template for easy template management. The description can contain up to 200 characters.



c) Click OK.

To export the parameters of the RDS instance, follow these steps:

Click Export Parameters. The parameters are exported as a file in TXT format to your computer. The file contains all the parameters and their settings of the RDS instance.

To import the parameters of the RDS instance, follow these steps:

a) After you finish modifying the exported parameter file, click Import Parameters and copy the content of the parameter file to the Import Parameters dialog box.

b) Click OK.

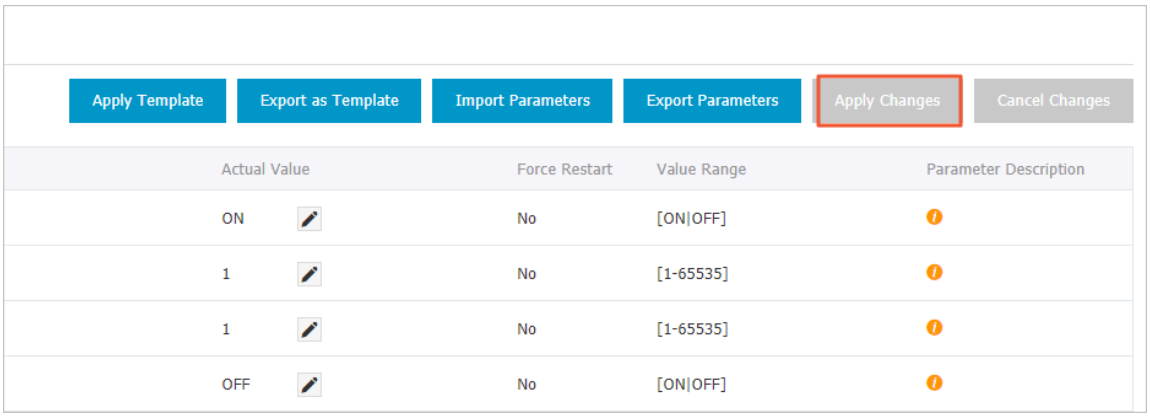
c) In the upper-right corner, click Apply Changes.

Note:

- If the new parameter settings take effect only after the RDS instance is restarted, the system asks you whether you want to restart the RDS

instance. If so, we recommend that you reconfigure parameters during off-peak hours and make sure that your application can automatically reconnect to the RDS instance.

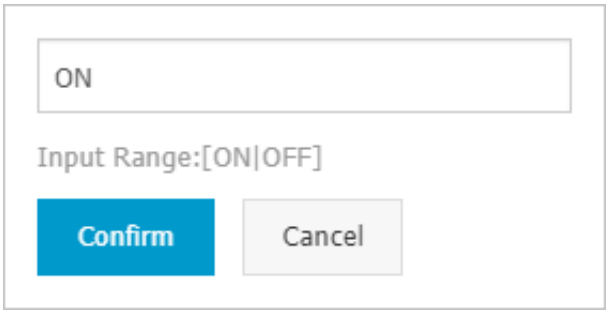
- Before the new parameter settings take effect, you can click Cancel Changes to cancel the parameter reconfiguration.



To reconfigure only one parameter of the RDS instance, follow these steps:

a) On the Modifiable Parameters tab, find the parameter you want to reconfigure, and in the Actual Value column click .

b) In the displayed dialog box, enter a new value within the value range.



c) Click Confirm.

d) In the upper-right corner, click Apply Parameters.

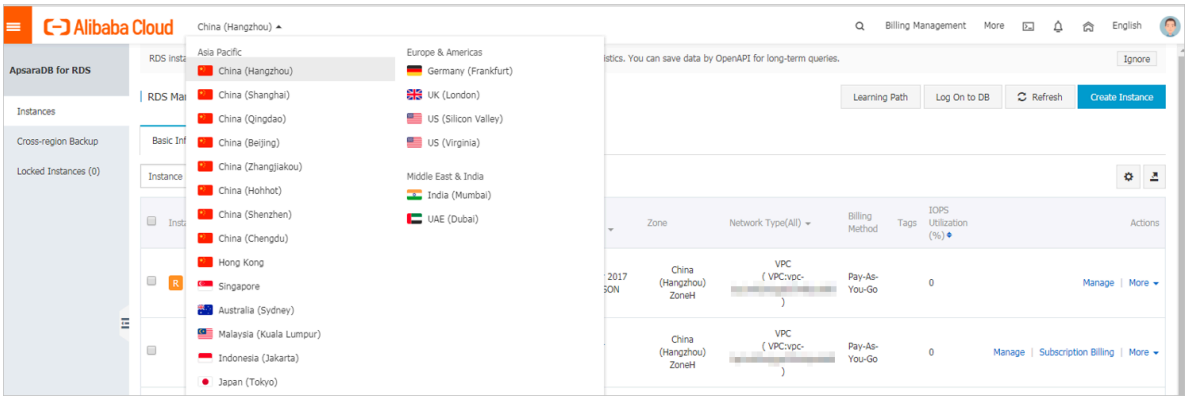
Note:

- If the new parameter settings take effect only after the RDS instance is restarted, the system asks you whether you want to restart the RDS instance. If so, we recommend that you reconfigure parameters during off-peak hours and make sure that your application can automatically reconnect to the RDS instance.

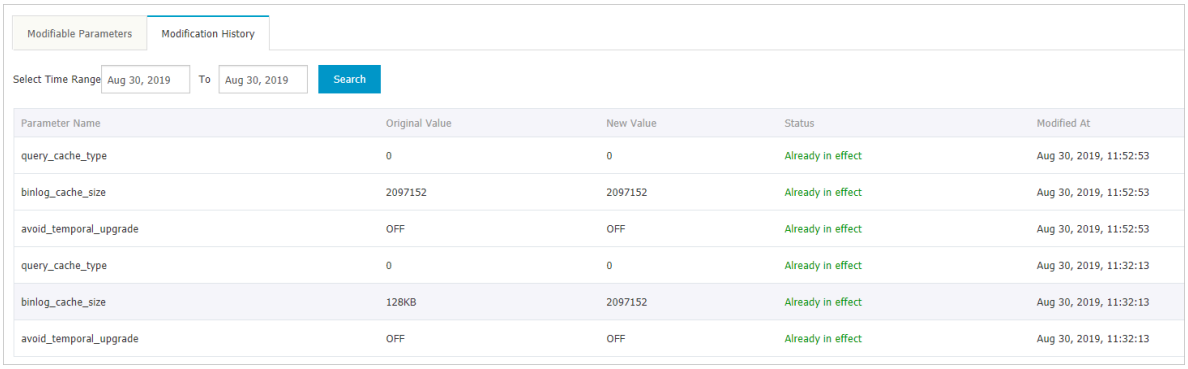
- **Before the new parameter settings take effect, you can click Cancel Changes to cancel the parameter reconfiguration.**

View the parameter reconfiguration history

1. **Log on to the *RDS console*.**
2. **Select the target region.**



3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Parameters.**
5. **Click the Modification History tab.**
6. **Select a time range and click Search.**



6.11 Operate parameter templates

This topic describes how to create, manage, and use parameter templates. You can use a parameter template to manage RDS for MySQL instances in batches.

Prerequisites

The instance version is one of the following:

- **MySQL 8.0**
- **MySQL 5.7**

- MySQL 5.6

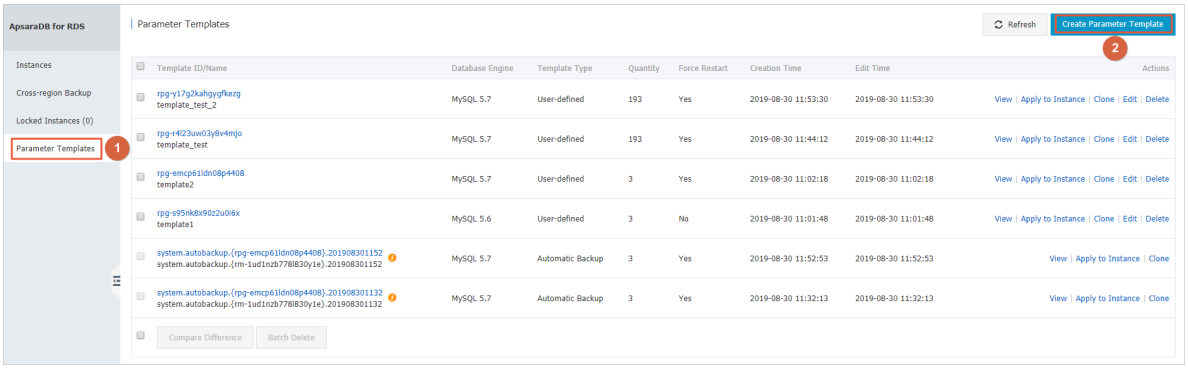
Context

If you only want to reconfigure only one parameter of an RDS instance, see *Reconfigure parameters for an RDS for MySQL instance*.

Create a parameter template

1. Log on to the *RDS console*.
2. In the upper-left corner, select the region where the target RDS instance is located.

3. In the left-side navigation pane, click Parameter Templates, and in the upper-right corner click Create Parameter Template.



4. Set the following parameters.

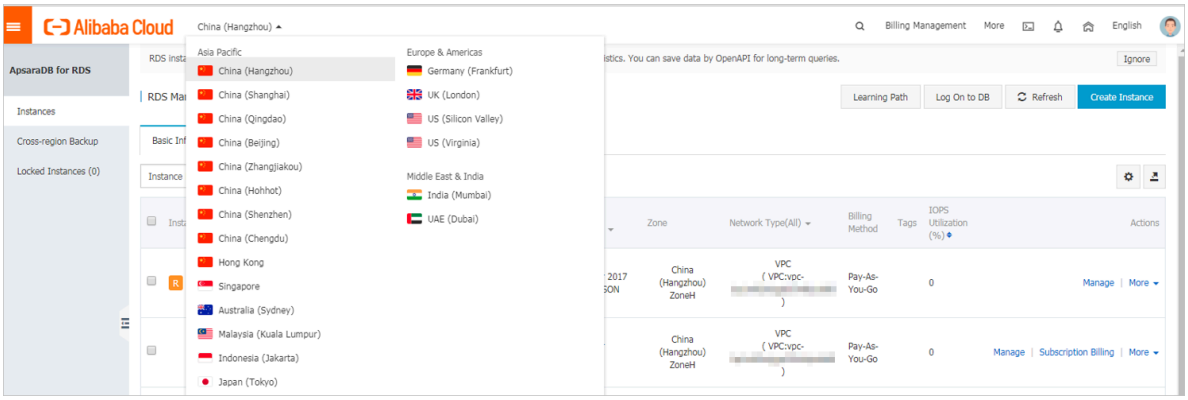
Parameter	Description
Template Name	The name of the parameter template. The template name must start with a letter and can contain 8 to 64 characters including letters, digits, decimal points (.), and underscores (_).
Database Engine	Only one option is available: MySQL.
Version	Two options are available: MySQL 5.6 and MySQL 5.7.

Parameter	Description
Description	The description of the parameter template for easy template management. The description can contain up to 200 characters.
Add Parameter	If you click Add Parameter, a modifiable parameter entry is displayed for you to specify the parameter name and value. Additionally, you can view the other information such as the value range and default value. **Note:** · To configure one more parameter, you can click Add Parameter again. · To delete a parameter, you can click Delete for the parameter entry.
Import	After modifying an exported parameter template, you can click Import to paste this parameter template. For more information about parameter template export, see *Reconfigure parameters for an RDS for MySQL instance*.

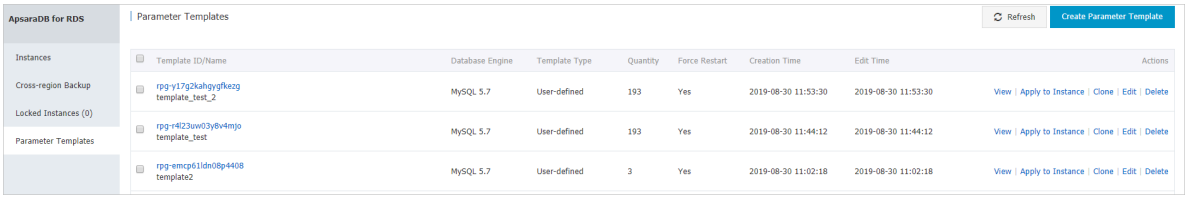
5. Click Confirm.

Manage parameter templates

1. Log on to the *RDS console*.
2. In the upper-left corner, select the region where the target RDS instance is located.



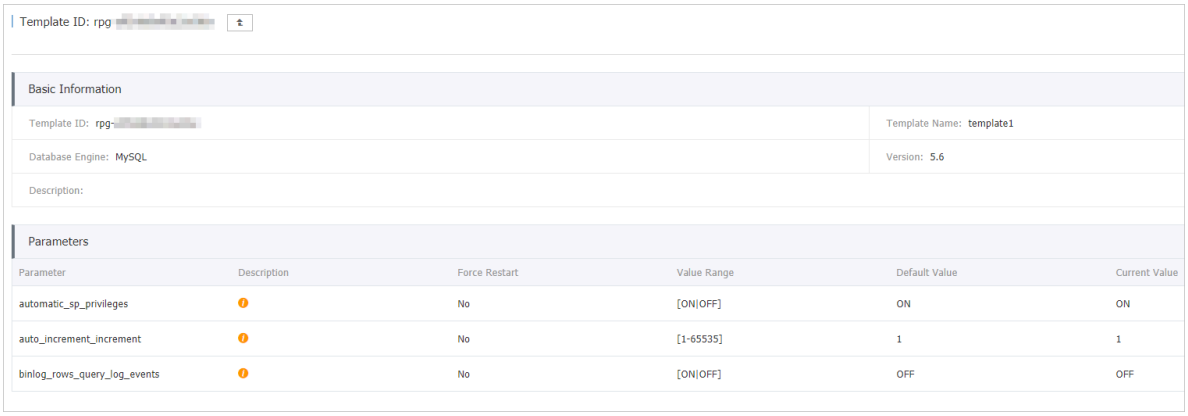
3. the left-side navigation pane, click Parameter Templates.



4. Perform operations on parameter templates as needed.

   To view a parameter template, follow these steps:

   Find the parameter template and in the Actions column click View. Then, you can view the basic information and parameter details of the template.



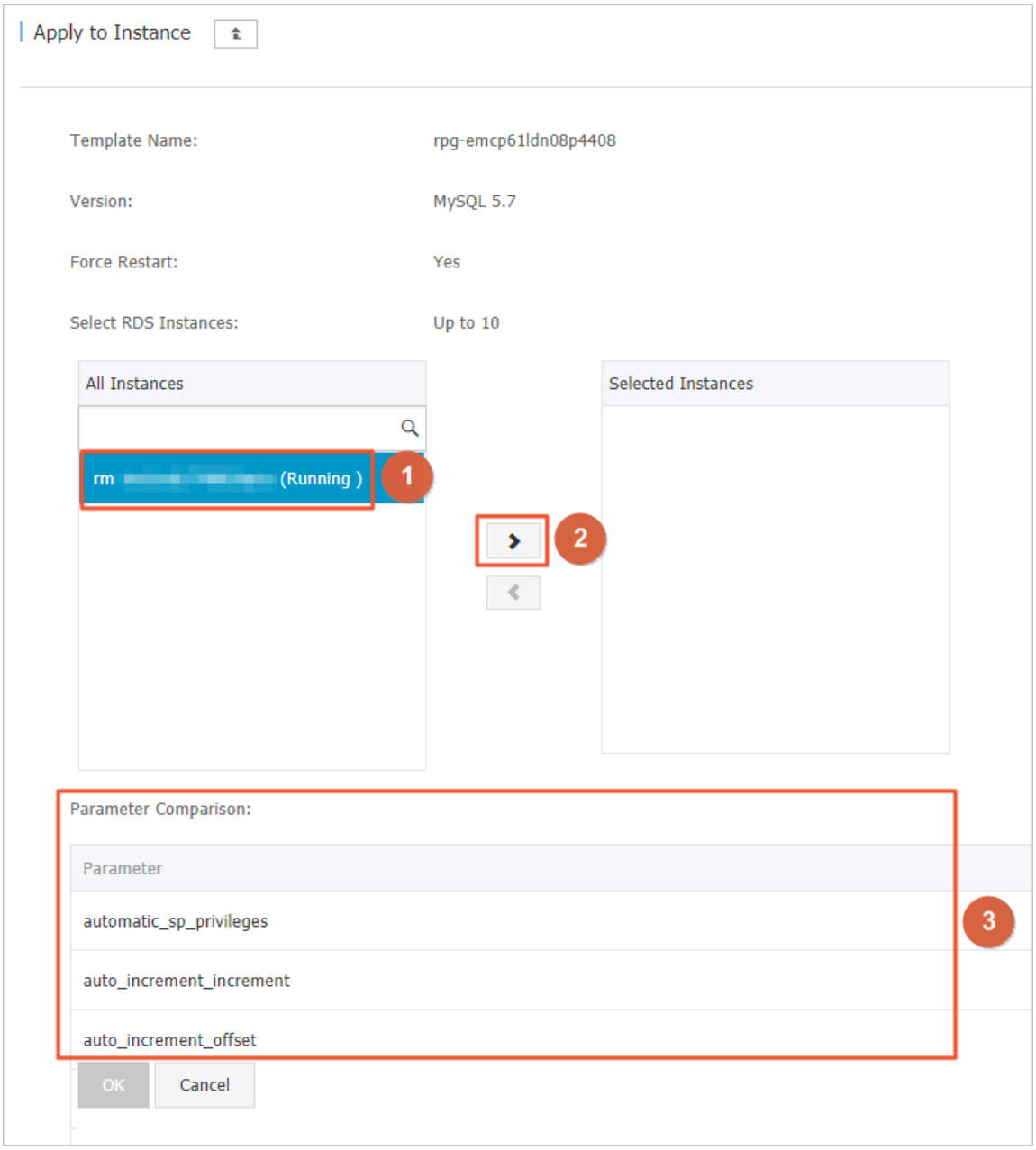
   To apply a parameter template to one or more RDS instances, follow these steps:

Note:

Before you apply a parameter template to more than one RDS instance, you must make sure that these parameters are suitable to these instances.

a) **Find the parameter template and in the Actions column click Apply to Instance.**

b) **Select RDS instances from the All Instances section, and click** > **to move them to the Selected Instances section. Then, check the parameter comparison information.**



c) **Click OK.**

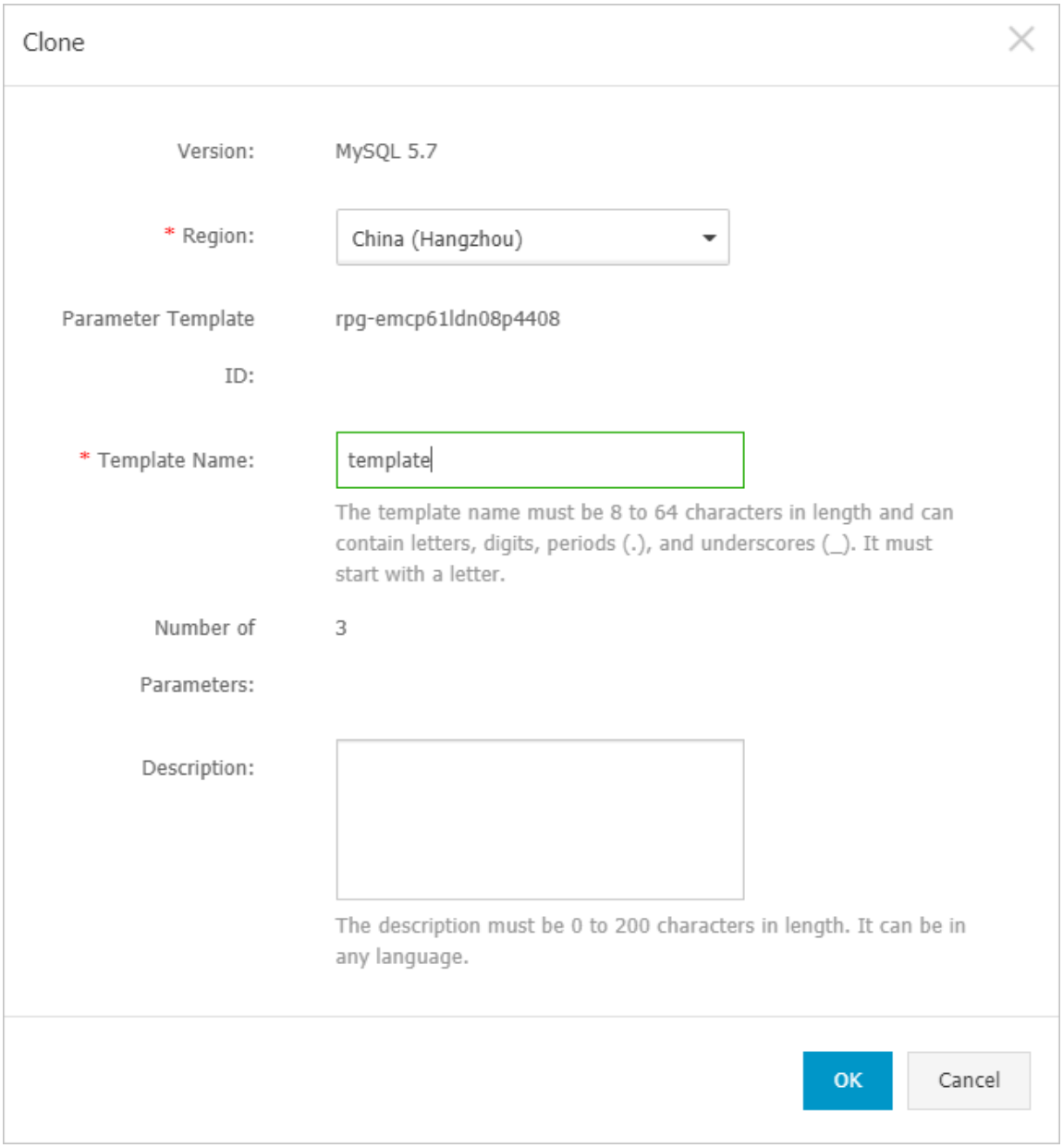
To replicate a parameter template from one region to another, follow these steps:

a) Find the parameter template and in the Actions column click Clone.

b) Set the following parameters

Parameter	Description
Region	The destination region to which the parameter template is to be replicated.
Parameter Template	The name of the parameter template. The template name must start with a letter and can contain 8 to 64 characters including letters, digits, decimal points (.), and underscores (_).

Parameter	Description
Description	The description of the parameter template for easy template management. The description can contain up to 200 characters.



c) Click OK.

To modify a parameter template, follow these steps:

a) Find the parameter template and in the Actions column click Edit. On the displayed page, modify the parameter template as needed.

b) Click Edit.

To delete a parameter template, follow these steps:

Find the parameter template and in the Actions column click Delete. In the displayed dialog box, click OK.

> **Note:**
> **Deleting a parameter template does not affect the RDS instances to which the template is applied.**

6.12 Instance recycle bin

This topic describes the instance recycle bin and the related operations. You can unlock, re-create, or destroy RDS instances in the recycle bin.

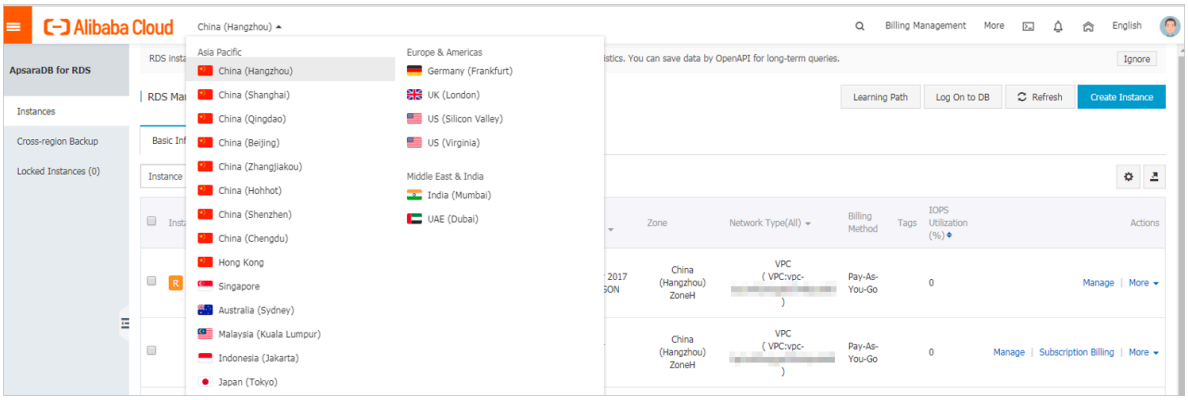
Unlock an RDS instance that has overdue payments

When an RDS instance is locked due to overdue payments, you can go to the recycle bin to renew and unlock it.

Unlock an RDS instance that has expired

When an RDS instance is locked due to expiration, you can go to the recycle bin to renew and unlock it.

1. **Log on to the *RDS console*.**
2. **Select the target region.**



3. **In the left-side navigation pane, click Locked Instances.**
4. **Find the locked instance and click Unlock to renew the instance.**

   The instance is unlocked after renewal.

Re-create an RDS instance

After an RDS instance expires, it is released on the eighth day and its data keeps stored in the system for another eight days. Within the eight days, you can restore

the instance data to a new instance. On the 16th day counting from the day when the RDS instance expired, the instance data is deleted from the system and cannot be retrieved.

1. Log on to the *RDS console*.
2. Select the target region.

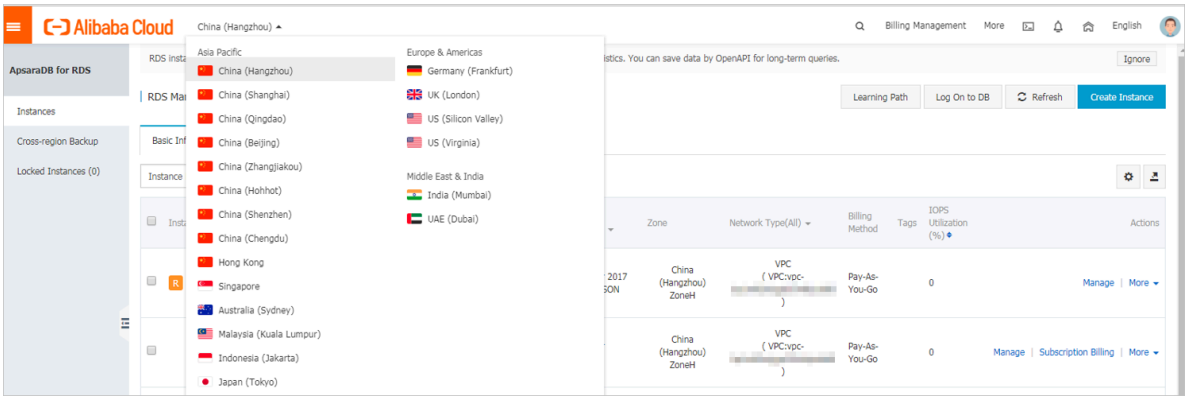
3. In the left-side navigation pane, click Locked Instances.
4. Find the target RDS instance and click the button for re-creating an RDS instance.

   By default, the system creates an RDS instance that has the same specifications and resides in the same zone as the released RDS instance. You can select other zones and specifications as needed.

Destroy an RDS instance

When an RDS instance is locked due to expiration or overdue payments, you can go to the recycle bin to destroy it.

1. Log on to the *RDS console*.
2. Select the target region.



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

4. **Find the instance and click Destroy.**

7 Read-only instances and read/write splitting

7.1 Read-only instances overview

This topic provides an overview of read-only instances in RDS for MySQL. You can create one or more read-only instances to support a massive number of read requests and increase the throughput of applications.

Background information

An RDS for MySQL read-only instance is a read-only copy of the corresponding master instance. Changes to the master instance are automatically replicated to all corresponding read-only instances. You can set the *replication delay* in each read-only instance.

Note:

- **Currently the following MySQL versions and RDS editions support read-only instances:**
  - **MySQL 8.0 High-availability Edition (with local SSDs)**
  - **MySQL 5.7 High-availability Edition (with local SSDs)**
  - **MySQL 5.6**
- **Each read-only instance adopts a single-node architecture (that is, without slave nodes).**

Pricing

Read-only instances are billed by using the pay-as-you-go billing method. For more information, see *ApsaraDB RDS for MySQL pricing*.

Features

Read-only instances in RDS for MySQL offer the following features:

- **A read-only instance is located in the same or different zones from its corresponding master instance.**
- **The specifications of a read-only instance do not need to be the same as that of the corresponding master instance and can be changed at any time. We recommend that the specifications of a read-only instance be greater than or**

equal to that of the corresponding master instance. If the specifications of a read-only instance is lower than that of the corresponding master instance, the read-only instance may suffer from high delay or load.

- The *network type* of a read-only instance can be different from that of its corresponding master instance.
- No account or database maintenance is required for a read-only instance. Both accounts and databases are replicated from the corresponding master instance.
- When you create a read-only instance, the system automatically replicates the whitelists of the corresponding master instance. After a read-only instance is created, its whitelists are independent of the whitelists of the corresponding master instance. If you want to change the whitelists of the read-only instance, see *Configure a whitelist for an RDS for MySQL instance*.
- Read-only instances support monitoring and alerting. Specifically, up to 20 performance metrics, including the disk capacity, IOPS, number of connections, CPU utilization, and network traffic.

Limits

- The number of read-only instances that a master instance can have varies depending on its memory size.

Database type	Memory	Max number of read-only instances
MySQL	⩾ 64 GB	10
	< 64 GB	5

- Read-only instances do not support backup settings or manual backups.
- Read-only instances do not support the creation of temporary instances through backup files or backups, or support the overwriting of instances by using backup sets.
- After a read-only instance is created, the master instance does not support data recovery through the direct overwriting of instances by using backup sets.
- You cannot migrate data to read-only instances.
- You cannot create or delete databases for read-only instances.
- You cannot create or delete accounts for read-only instances.
- You cannot authorize accounts or change account passwords for read-only instances.

FAQ

Can the accounts on the master instance be used on the read-only instances attached with it?

Accounts on the master instance are replicated to the read-only instances. You can use the accounts to read data from the read-only instances but cannot write data into the read-only instances.

7.2 Create an RDS for MySQL read-only instance

This topic describes how to create an RDS for MySQL read-only instance. You can create read-only instances to process massive read requests sent to the database and increase the application throughput. A read-only instance is a read-only copy of the master instance. Changes to the master instance are also automatically synchronized to all relevant read-only instances.

For more information, see *Read-only instances overview*.

Prerequisites

The DB engine version and RDS edition are as follows:

- **MySQL 8.0 High-availability Edition (with local SSDs)**
- **MySQL 5.7 High-availability Edition (with local SSDs)**
- **MySQL 5.6**

Precautions

- **You can create read-only instances in your master RDS instance. However, you cannot switch an existing instance to a read-only instance.**
- **Creating a read-only instance does not affect your master RDS instance because the read-only instance replicates data from the slave instance.**
- **A read-only instance does not inherit the parameter settings of your master RDS instance. Instead, default parameter settings are generated. You can modify the parameter settings of a read-only instance on the RDS console.**
- **The quantity of read-only instances is as follows.**

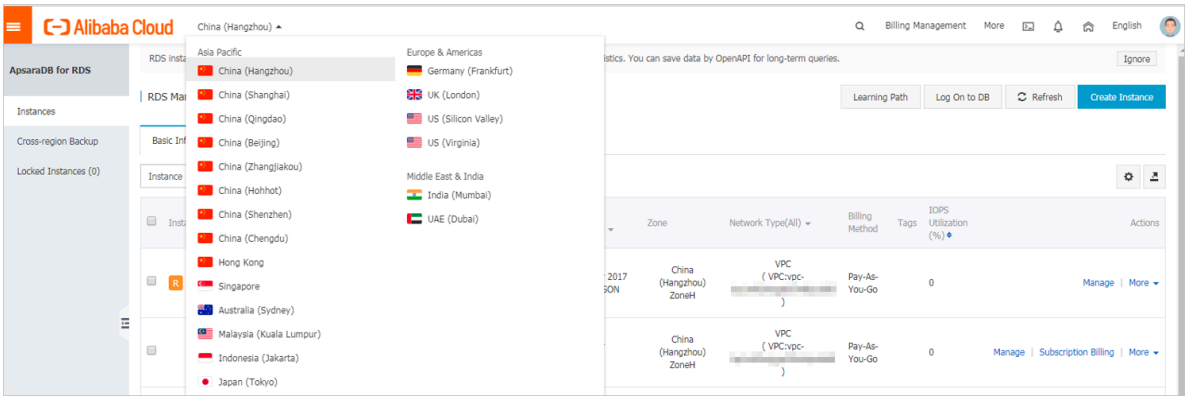
Database type	Memory	Max number of read-only instances
MySQL	⩾ 64 GB	10

Database type	Memory	Max number of read-only instances
	< 64 GB	5

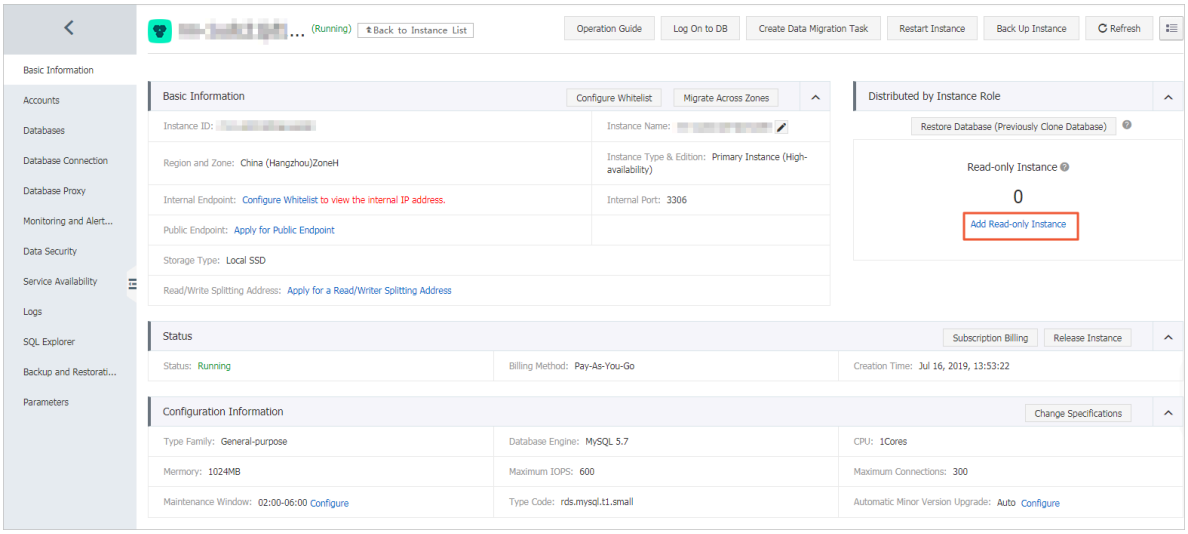
- **A read-only instance is charged by using the pay-as-you-go billing method. Specifically, the fees for a read-only instance are deducted once per hour depending on the instance specifications. For more information, see the "Read-Only Instances" part at *ApsaraDB RDS for MySQL pricing*.**

Create a read-only instance

1. **Log on to the *RDS console*.**
2. **Select the target region.**



3. **Find the target RDS instance and click the instance ID.**
4. **Click Add Read-only Instance.**



5. On the purchase page, choose the configuration of the read-only instance, and then click Buy Now.

   Note:

   - We recommend that the read-only instance and its corresponding master instance be located in the same VPC.
   - To guarantee sufficient I/O for data synchronization, we recommend that the configuration of the read-only instance (specifically, the memory) is greater than or equal to that of the master instance.
   - We recommend that you purchase multiple read-only instances based on your business needs to improve availability.

6. On the Order Confirmation page, read and confirm you agree to Terms of Service, Service Level Agreement, and Terms of Use by selecting the checkbox, confirm the order details, and click Pay Now.

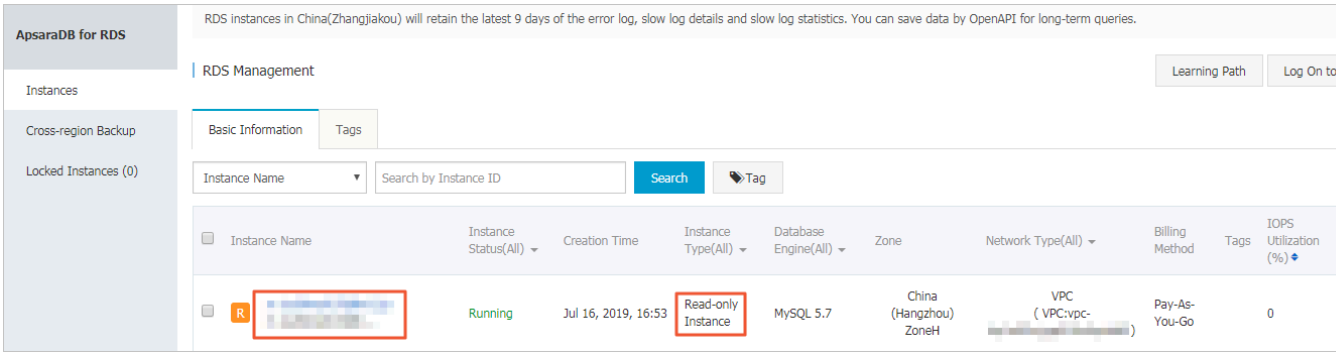
   The instance creation takes a few minutes.

View a read-only instance

To view a read-only instance in the instance list, follow these steps:

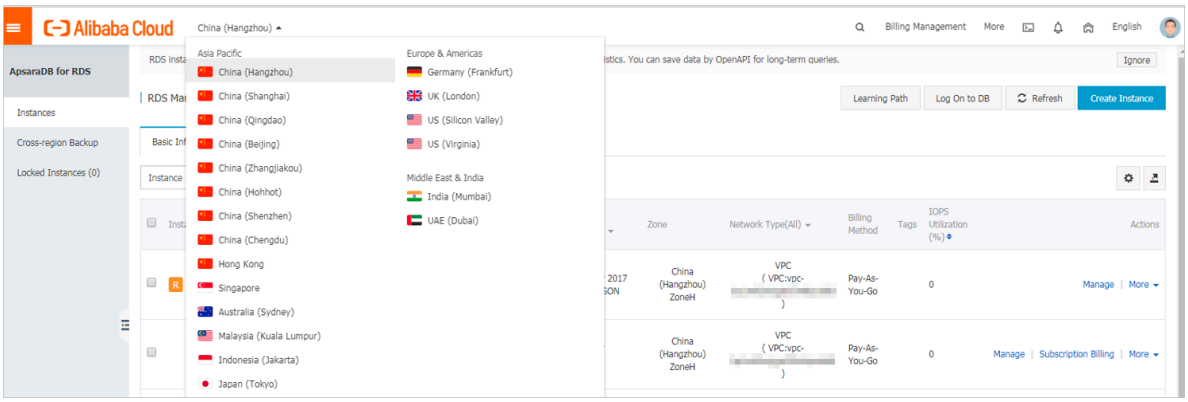
1. Log on to the *RDS console*.
2. Select the target region.

3. Find the target read-only instance and click the instance ID.

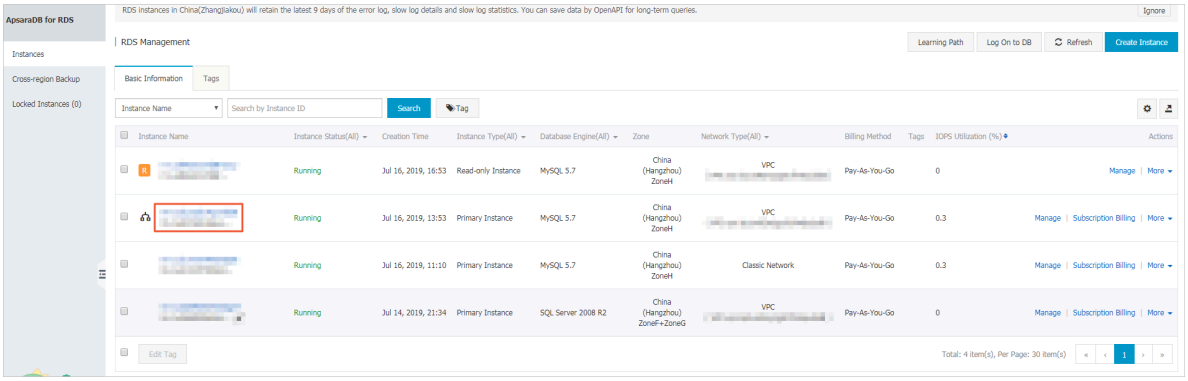


To view a read-only instance on the Basic Information page for the master instance, follow these steps:

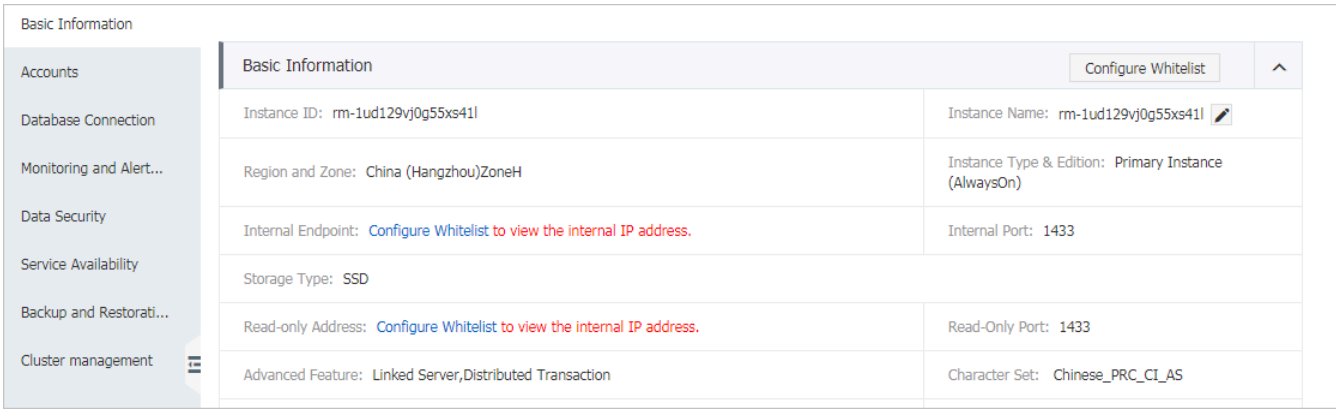
1. Log on to the *RDS console*.
2. Select the target region.



3. Find the master instance and click the instance ID.

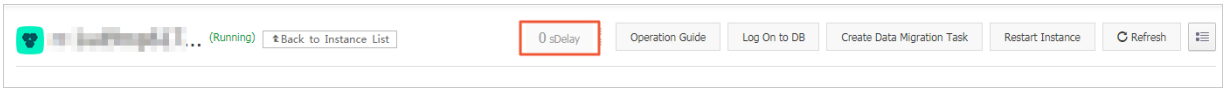


4. On the Basic Information page, move the pointer over the number below Read-only Instance and click the ID of the read-only instance.



View the delay time of a read-only instance

When a read-only instance replicates data from the master instance, the read-only instance may lag behind the master instance by a small amount of time. You can view the delay on the Basic Information page of the read-only instance.



APIs

API	Description
#unique_53	Used to create an RDS read-only instance.

7.3 Set a replication delay for an RDS for MySQL read-only instance

This topic describes how to set a replication delay for an RDS for MySQL read-only instance.

Context

You can set a replication delay for a read-only instance of an RDS for MySQL instance. Then the read-only instance synchronizes data from the master instance at the specified delay.

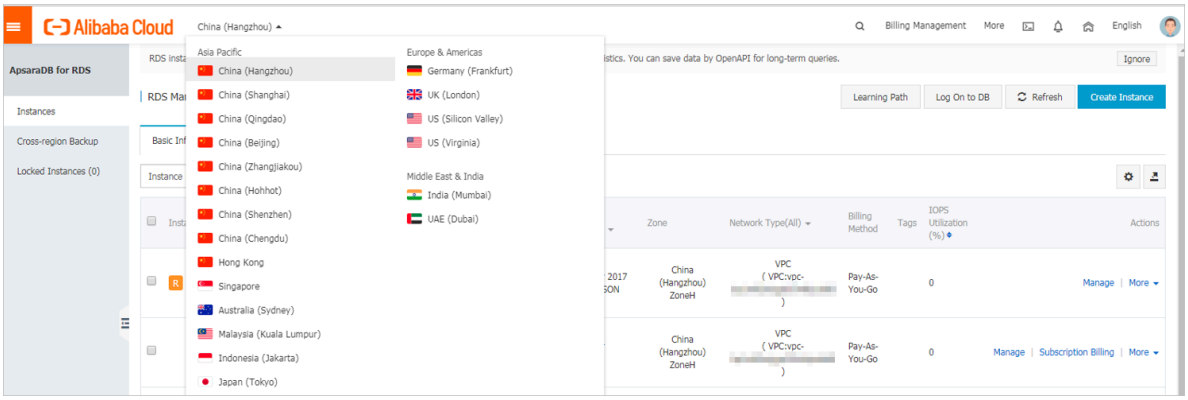
For more information about the related APIs, see *#unique_114*.

Limits

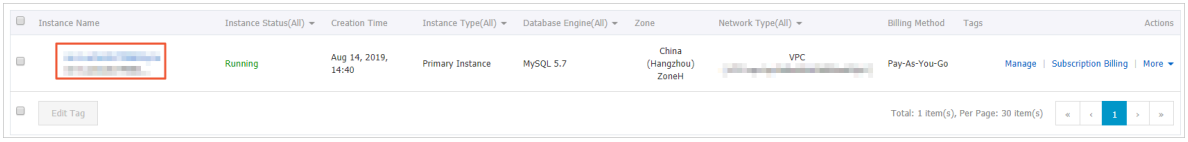
- If a replication delay is set for a read-only instance, you cannot add the read-only instance to *read/write splitting*.
- If a read-only instance is in *read/write splitting*, you cannot set a replication delay for it.

Procedure

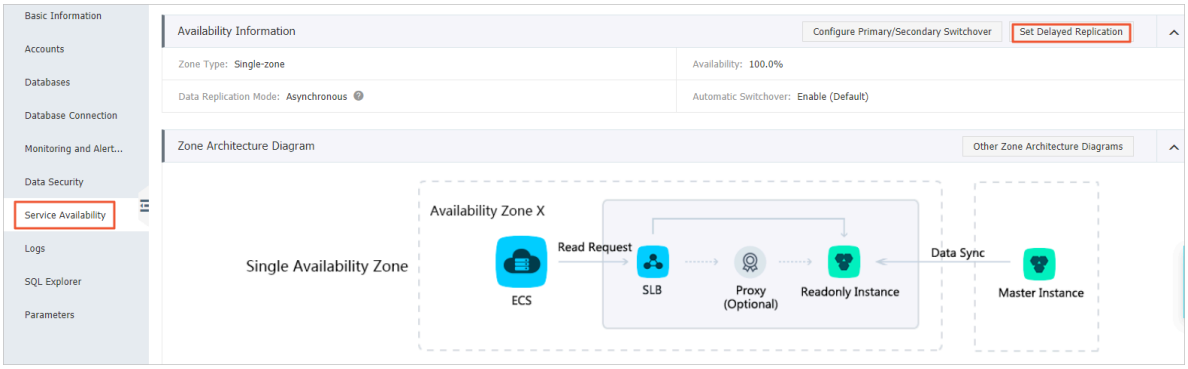
1. Log on to the *RDS console*.
2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.



4. In the left-side navigation pane, click Service Availability.
5. Click Set Delayed Replication.



6. In the displayed dialog box, set Delay and click OK.

Note:

The delay is measured in the unit of second. The default delay is 0 seconds, indicating that the master instance sends operation log data to the read-only instance immediately after completing an operation and the read-only instance performs the operation immediately after receiving the operation log data.

7.4 Introduction to read/write splitting

This topic introduces the read/write splitting function. This function enables RDS to distribute read and write requests through a read/write splitting address.

Functions

When read/write splitting is enabled, three types of connection addresses are available in the instances:

- **The connection address of the master instance: The Internet and intranet IP addresses can coexist.**
- **The connection address of the read-only instance: The Internet and intranet IP addresses can coexist.**
- **The read/write splitting address: The Internet and intranet IP addresses cannot coexist. An intranet IP address is generated by default. If you must use the read/write splitting Internet IP address, you can switch the IP address. For more information, see** *Change the network type of the read/write splitting address for an RDS for MySQL instance***.**

The master instance and read-only instance require independent connection addresses. Currently, instance connection addresses are configured automatically in applications to split read and write operations.

With this function, an extra read/write splitting address is provided to associate the master instance with all its read-only instances, achieving an automatic read/write splitting link. Applications can perform read and write operations with a single connection address. Write requests are routed automatically to the master instance and read requests are routed to each read-only instance by user-defined weights. You can scale up the processing capacity of the system by adding more read-only instances without any change to applications.

The following figure shows different types of connection addresses.

Benefits

- **Facilitates maintenance with a single read/write splitting address.**

  The master and read-only instances each have an independent connection address. You must configure each of these addresses in your application so that write requests are sent to the master instance and read requests to the read-only instances.

  The read/write splitting function provides an additional address called read/write splitting address. You can connect to this address to perform read and write operations on the master and read-only instances, with read and write requests automatically distributed. Therefore, maintenance costs are reduced.

  Additionally, you can increase the processing capability of your DB system by adding read-only instances without making any changes to your application.
- **Improves performance with support for the highly secure link.**

  For users who build a proxy layer to implement read/write splitting on the cloud, data has to go through multiple components for statement parsing and forwarding before it reaches the database, significantly increasing the response latency. RDS read/write splitting can be directly set in the existing highly secure link without time consumption by any other components, which reduces the latency and improves the processing rate.
- **Applies to various scenarios with customizable weights and thresholds.**

  RDS read/write splitting can be used to set read request weights for master and read-only instances and latency thresholds for read-only instances.
- **Enhances database availability with instance health checks.**

  RDS read/write splitting performs health check automatically for all instances in the distribution system. If any instance fails or its latency exceeds the threshold, RDS automatically removes the instance out of the distribution system (while marking it as unavailable and stopping allocating read requests to it) and allocates read and write requests to the remaining healthy instances by the predefined weights. In this way, applications still run properly even if any single-node read-only instance fails. After the instance resumes, RDS automatically reclaims it into the request distribution system.

  Note:

To prevent single node failures, we recommend that you create at least two read-only instances for each master instance if you are using read/write splitting.

- **Reduces resource and maintenance costs with free services.**

  RDS provides free read/write splitting function for all users.

Restrictions

- **Currently, the following commands or functions cannot be forwarded to a read-only instance:**
  - **The `stmt prepare sql` command is automatically executed on the master instance.**
  - **The `stmt prepare command` command cannot be forwarded to a read-only instance before execution of `stmt close`.**
  - **The environment configuration variables `set global`, `set user`, and `set once` are automatically executed on the master instance.**
- **The following commands or functions are not supported currently:**
  - **SSL encryption**
  - **Compression protocols**
  - **com_dump_table and com_change_user protocols**
  - `kill connection [query]`
  - `change user`
- **The execution result is random for the following commands:**

  The `show processlist`, `show master status`, and `com_process_info` commands return results according to the instance connected during execution.
- **All transactions are routed to the master database.**
- **Read/write splitting does not guarantee consistency of non-transactional reads. If you require such consistency, add hints to route query requests to the master database or encapsulate query requests into transactions.**
- **The `LAST_INSERT_ID()` function is not supported. To use this function, add `hint : /*FORCE_MASTER*/, eg:/*FORCE_MASTER*/ SELECTLAST_INSERT_ID();` to the request.**

Reference

FAQ for read/write splitting

7.5 Enable read/write splitting for an RDS for MySQL instance

This topic describes how to enable the read/write splitting function for an RDS for MySQL instance. When this function is enabled, your application connects to a read/write splitting module through a read/write splitting address. The read/write splitting module distributes write requests to the master instance, and distributes read requests to the read-only instances based on the preset read weights.

Prerequisites

- **The RDS instance is a master instance.**
- **The DB engine version and edition of the RDS instance are as follows:**
  - **MySQL 5.7 High-availability Edition (with local SSDs)**
  - **MySQL 5.6**

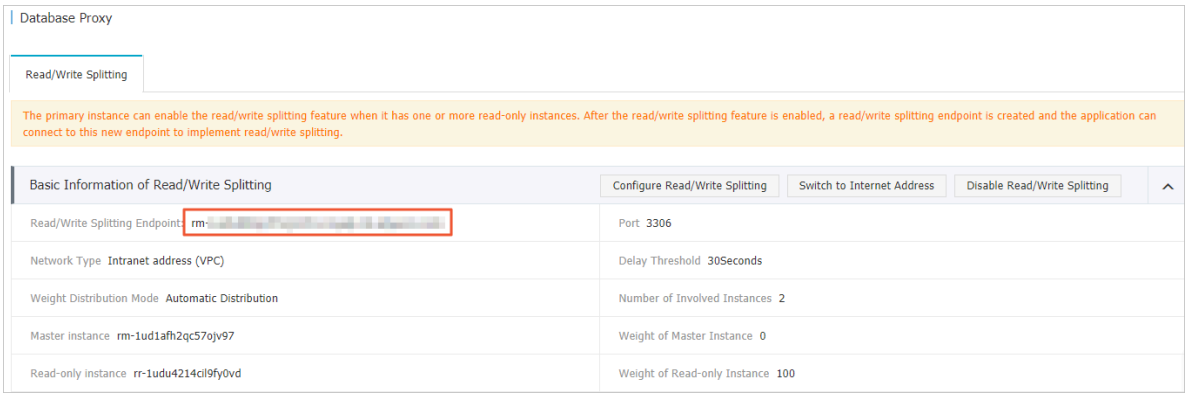
  Note:
  MySQL 8.0 supports read-only instances but does not support read/write splitting.
- **The RDS instance belongs to one of the following regions: China (Hangzhou), China (Shanghai), China (Qingdao), China (Beijing), China (Zhangjiakou), China (Hohhot), China (Shenzhen), China (Hong Kong), Singapore, Australia (Sydney), US (Virginia), US (Silicon Valley), Germany (Frankfurt), and UK (London).**
- **The RDS instance has at least one read-only instance. For more information about how to create a read-only instance, see** *Create an RDS for MySQL read-only instance*.
- **The** *replication latency* **is not enabled for the read-only instances of the RDS instance.**

Precautions

- **If it is the first time that you enable the read/write splitting function, the system automatically upgrades the backend administration systems of the master and read-only instances to the latest version to guarantee service availability. When the read/write splitting function is being enabled, the master instance is disconnected for 30 seconds or less. Additionally, the read-only instances is inaccessible during the whole restart process. We recommend that you enable the read/write splitting function during off-peak hours and make sure that your application can automatically reconnect to the RDS instance.**

- **If you have restarted or changed the specifications at least once for the master instance and its associated read-only instances after March 8, 2017, the backend administration systems of these instances have been automatically upgraded to the latest version. In such case, when you enable the read/write splitting function, the system does not restart the RDS instance or generate a transient disconnection.**
- **The generated read/write splitting address is fixed. It does not change even when you enable and disable the read/write splitting function for multiple times. Therefore, you do not need to change the configuration data on your application frequently. This reduces maintenance costs.**

> **Note:**
> **The read/write splitting address cannot be changed manually.**

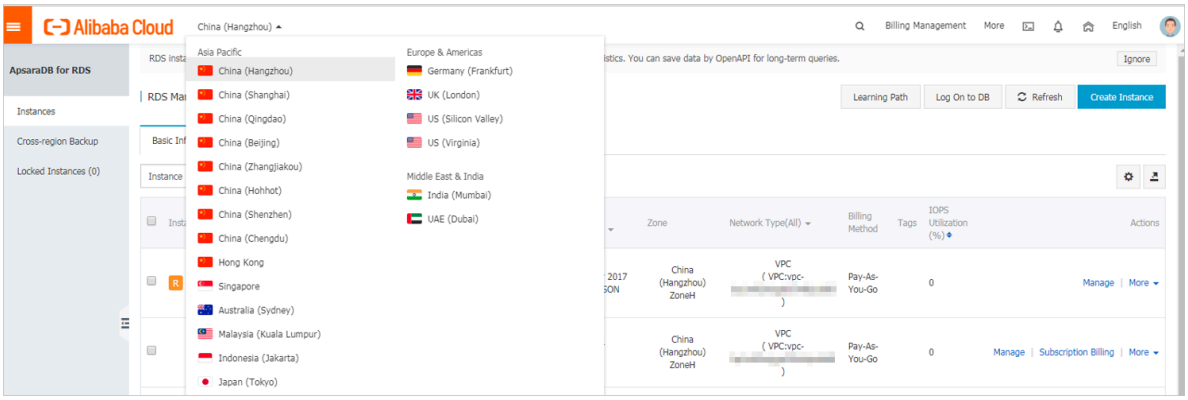


- **The read/write splitting function is free of charge. You pay only for the *read-only instances* you use.**

Procedure

1. **Log on to the *RDS console*.**

2. In the upper-left corner, select the region where the target RDS instance is located.

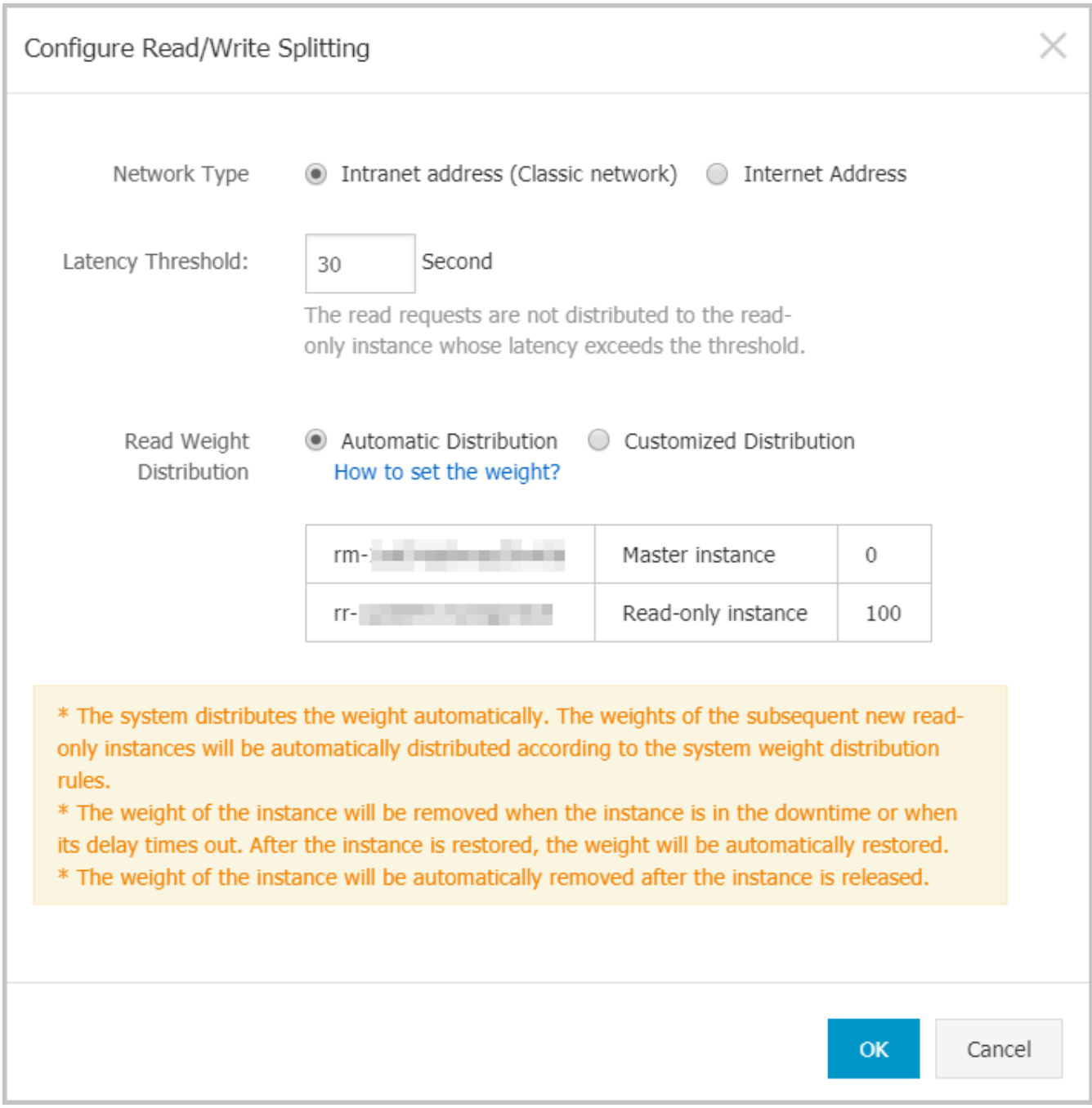


3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Database Connection or Database Proxy.
5. On the Read/Write Splitting tab, click Enable now.

Note:

- If theEnable now button is not displayed, you must enable the database proxy and create at least one read-only instance.
- If the master and read-only instances were created before March 8, 2017 and you have not restarted them or changed their specifications, enabling the read/write splitting function incurs a restart of the instances. In such case, you must click OK in the displayed dialog box to enable the read/write splitting function.

6. Set the following parameters.



Parameter	Description
Network Type	**The type of the read/write splitting address.** **· Intranet address** **- If the network type of the master instance is VPC, then the network type of the read/write splitting address is also VPC.** **- If the network type of the master instance is Classic Network, then the network type of the read/write splitting address is also Classic Network.** **· Internet Address: used to access the RDS instance from the Internet. The Internet is prone to fluctuations. Therefore, we recommend that you use an internal endpoint to connect to the RDS instance.**
Latency Threshold	**The maximum latency allowed for data synchronization between the master and read-only instances. When the data synchronization latency of a read-only instance exceeds this threshold, the system no longer distributes read requests to the read-only instance no matter how high the read weight of the read-only instance is.** **The latency threshold ranges from 0 seconds to 7,200 seconds.** **Due to limitations in the execution of SQL statements, a read-only instance may synchronize data from the master instance at a certain latency. We recommend that you set this parameter to a value greater than or equal to 30 seconds.**

Parameter	Description
Read Weight Distribution	A higher read weight indicates more read requests to process. For example, the master instance has three read-only instances, and their read weights are 0, 100, 200, and 200, respectively. Then the master instance does not process read requests (write requests are still automatically sent to the master instance), and the three read-only instances process read requests at the 1:2:2 ratio. · Automatic Distribution: The system automatically assigns a read weight to each instance based on the instance specifications. After a read-only instance is created, it is automatically added to the read/write splitting link according to the read weight assigned by the system. For more information, see *Rules of weight allocation by the system*. · Customized Distribution: You can manually set the read weight of each instance. The value ranges from 0 to 10000. After a read-only instance is created, its read weight is 0 by default. You must manually specify a read weight for the read-only instance. **Note:** You cannot specify a read weight for a read-only instance for which a *replication latency* is specified.

7. Click OK.

Note:
The instance status changes to Creating Network Connection. Wait until the instance status changes back to Running.

What to do next

You can change the configuration data on your application so that your application uses the read/write splitting address to connect to the RDS instance.

Note:
You can obtain the read/write splitting address on the Basic Information page or on the Read/Write Splitting tab of the Database Connection page.

FAQ

- *Modify the latency threshold and read weights*
- *Change the network type of the read/write splitting address for an RDS for MySQL instance*

- *Monitor the read/write splitting performance of an RDS for MySQL instance*
- *Rules of weight allocation by the system*
- *FAQ for read/write splitting*

7.6 Modify the latency threshold and read weights

This topic describes how to modify the latency threshold and read weights when read/write splitting is enabled.

For more information, see *Table 7-1: Read/write splitting parameters*.

Procedure

1. **Log on to the** *RDS console*.
2. **In the upper-left corner, select the region where the target RDS instance is located.**

3. **Find target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Database Connection or Database Proxy.**
5. **Click the Read/Write Splitting tab.**

6. **In the Basic Information of Read/Write Splitting section, click Configure Read/Write Splitting. Then, set the following parameters.**

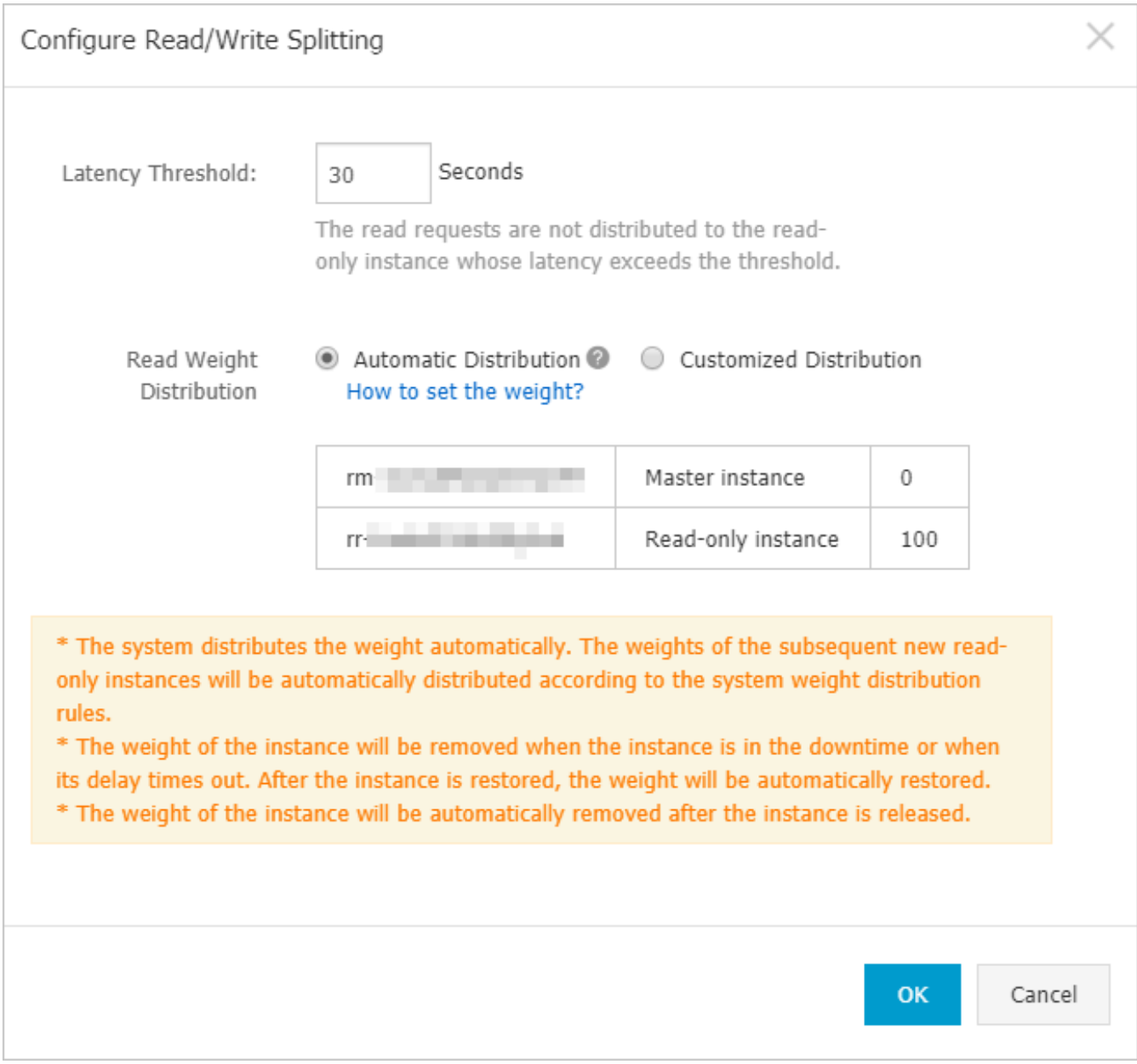


Table 7-1: Read/write splitting parameters

Parameter	Description
Latency Threshold	**The maximum latency allowed for data synchronization between the master and read-only instances. When the data synchronization latency of a read-only instance exceeds this threshold, the system no longer distributes read requests to the read-only instance no matter how high the read weight of the read-only instance is.** **The latency threshold ranges from 0 seconds to 7,200 seconds.** **Due to limitations in the execution of SQL statements, a read-only instance may synchronize data from the master instance at a certain latency. We recommend that you set this parameter to a value greater than or equal to 30 seconds.**
Read Weight Distribution	**A higher read weight indicates more read requests to process. For example, the master instance has three read-only instances, and their read weights are 0, 100, 200, and 0, respectively. Then the master instance does not process read requests (write requests are still automatically sent to the master instance), the first and second read-only instances process read requests at the 1:2 ratio, and the third read-only instance does not receive any read or write requests.** · **Automatic Distribution: The system automatically assigns a read weight to each instance based on the instance specifications. After a read-only instance is created, it is automatically added to the read/write splitting link according to the read weight assigned by the system. For more information, see** *Rules of weight allocation by the system*. · **Customized Distribution: You can manually set the read weight of each instance. The value ranges from `0` to `10000`. After a read-only instance is created, its read weight is `0` by default. You must manually specify a read weight for the read-only instance.** **Note:** · **If a read-only instance is deleted, then its read weight is deleted and the read weights of the other read-only instances remain unchanged.** · **You cannot specify a read weight for a read-only instance for which a***replication delay*.

7. **Click OK.**

Reference

FAQ for read/write splitting

APIs

API	Description
#unique_121	**Used to change the latency threshold and read weights of RDS instances.**

7.7 Change the network type of the read/write splitting address for an RDS for MySQL instance

This topic describes how to change the network type of the read/write splitting address for an RDS for MySQL instance. When you enable the read/write splitting function, you can use the internal or public endpoint of the master instance as the read/write splitting address.

If your application is deployed in the same VPC as the RDS instance, we recommend that you use the internal endpoint to guarantee data security and communication efficiency. If your application is deployed on your computer or in a different network from your RDS instance, you can use the public endpoint to establish a connection.

Prerequisites

The read/write splitting function is enabled. For more information, see *Enable read/write splitting for an RDS for MySQL instance***.**

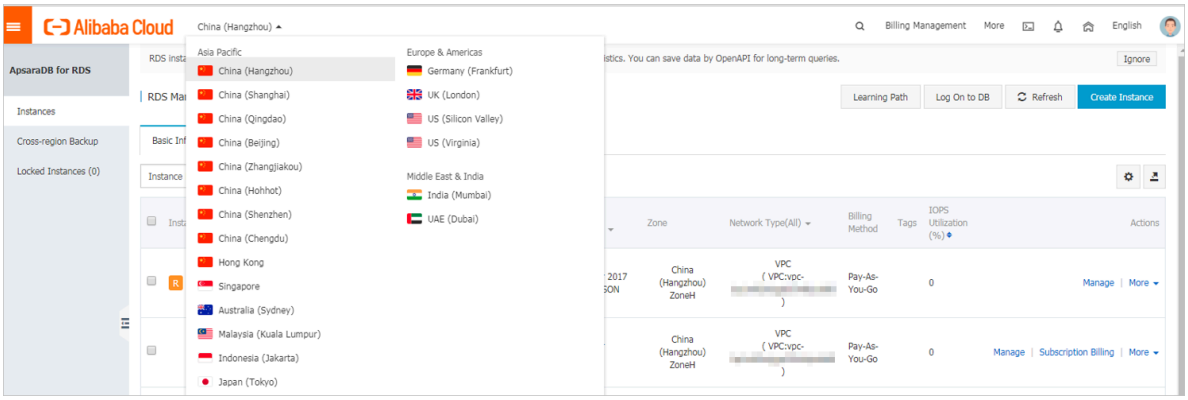
Precautions

When you change the network type of the read/write splitting address, the master instance experiences a transient disconnection for up to 30 seconds. To avoid the impact of the transient disconnection, we recommend that you change the network type during off-peak hours and make sure that your application can automatically reconnect to the RDS instance.

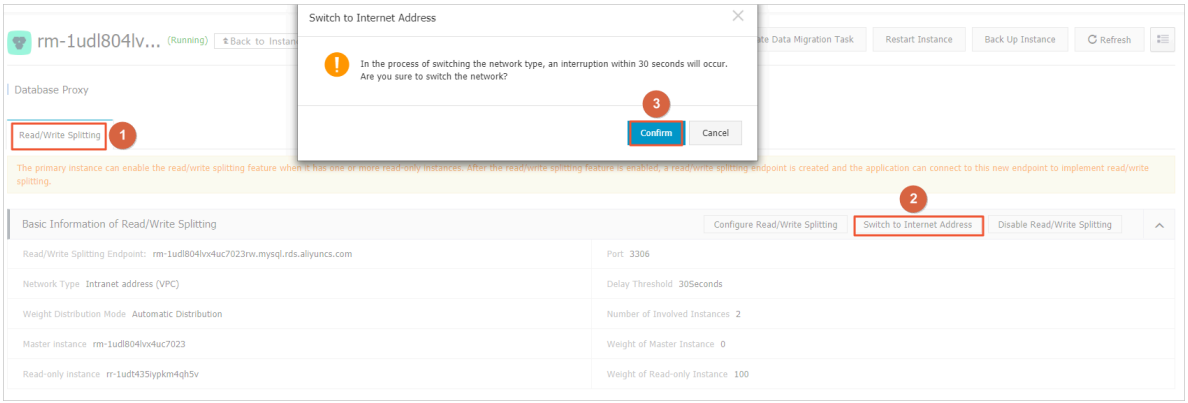
Procedure

1. **Log on to the** *RDS console***.**

2. **In the upper-left corner, select the region where the target RDS instance is located.**



3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Database Connection or Database Proxy.**
5. **Click the Read/Write Splitting tab.**
6. **Click Switch to Internet Address or Switch to Inner Address.**
7. **In the displayed dialog box, click Confirm.**



Reference

FAQ for read/write splitting

7.8 Disable read/write splitting for an RDS for MySQL instance

This topic describes how to disable read/write splitting for an RDS for MySQL instance.

Prerequisites

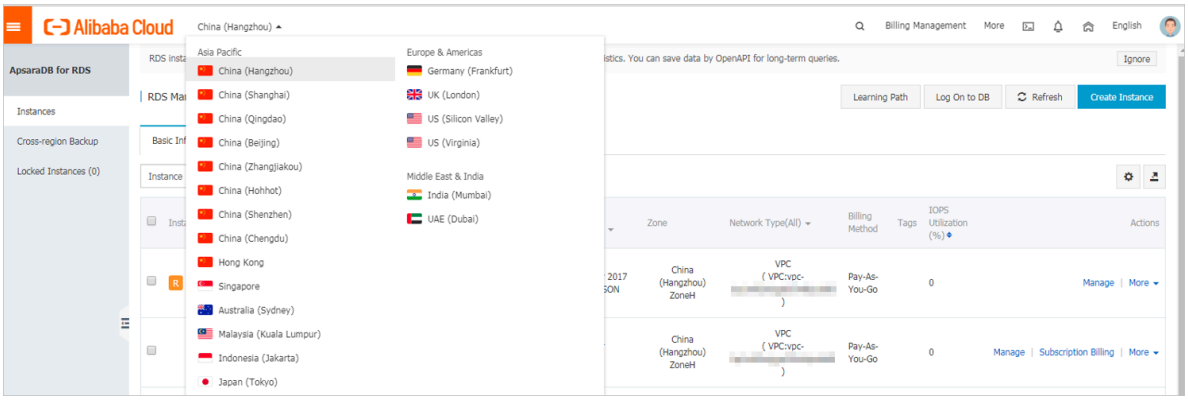
The read/write splitting function has been enabled for the RDS instance. For more information, see *Enable read/write splitting for an RDS for MySQL instance*.

Precautions

- **When the read/write splitting function is being disabled, the RDS instance is disconnected for 30 seconds or less. We recommend that you disable this function during off-peak hours and make sure that your application can automatically reconnect to the RDS instance.**
- **After the read/write splitting function is disabled, the read/write splitting address becomes invalid. Make sure that your application no longer uses the read/write splitting address to connect to the RDS instance.**

Procedure

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**



3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Database Connection or Database Proxy.**
5. **Click the Read/Write Splitting tab.**
6. **In the Basic Information of Read/Write Splitting section, click Disable Read/Write Splitting.**
7. **In the displayed dialog box, click Confirm.**

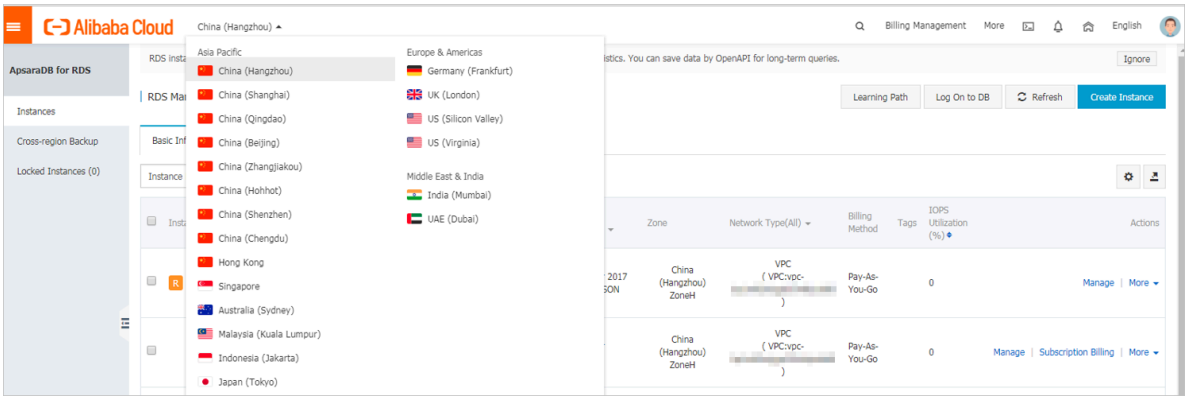
7.9 Monitor the read/write splitting performance of an RDS for MySQL instance

This topic describes how to monitor the read/write splitting performance of an RDS for MySQL instance in the RDS console.

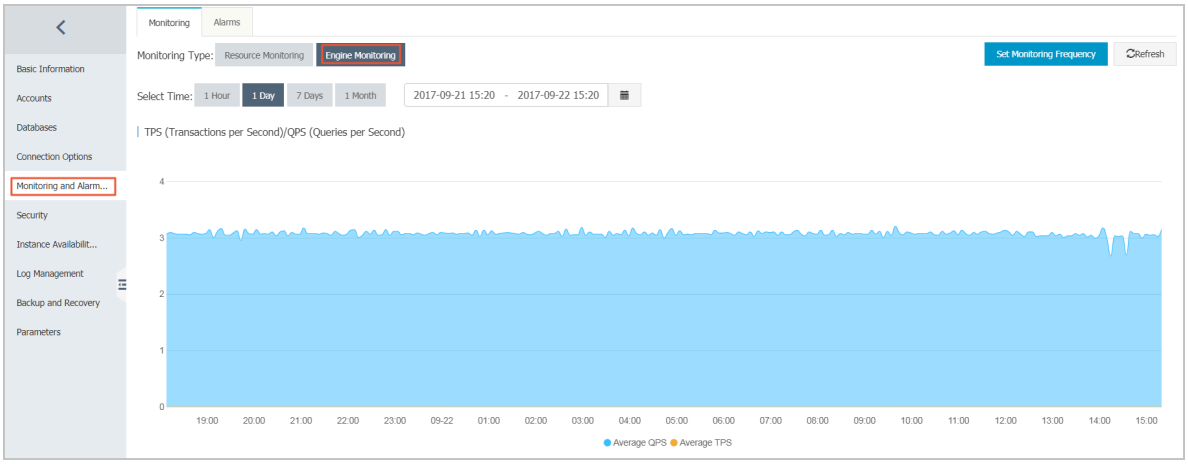
Procedure

1. **Log on to the *RDS console*.**

2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Monitoring and Alerts.
5. On the Monitoring tab, select Engine Monitoring.



6. View the TPS (Transactions per Second)/QPS (Queries per Second) chart to obtain the number of read and write operations on each of the master and read-only instances.

7.10 Verify the read/write splitting performance of an RDS for MySQL instance

This topic uses Sysbench 0.5, a MySQL stress testing tool, as an example to describe how to correctly configure parameters to verify the read/write splitting

performance of an RDS for MySQL instance. After read/write splitting is enabled, all transactions are routed to the master instance by default.

Prerequisites

- **The read/write splitting function is enabled. For detailed operations, see** *Enable read/write splitting for an RDS for MySQL instance*.
- **The Sysbench 0.5 is installed. For detailed operations, see the** *Sysbench documentation* .

Precautions

- **We recommend that a case with prepare or a transaction not be for testing the load balance performance of read/write splitting.**
- **Prevent the master/slave latency from exceeding the threshold set for the monitoring check due to high read stress.**
- **We recommend that you use the following Sysbench scripts to build a specific SQL statement as needed.**

```
function thread_init(thread_id)
      db_connect()
  end
  function event(thread_id)
      rs =  db_query("select 1")
  end
```

Set Sysbench parameters

A transaction is used by default to test the Sysbench oltp.lua script. If you use default parameters, all SQL statements are executed in the transaction and read-only SQL statements are routed to the master database for execution. Therefore, when the Sysbench is used to test read/write splitting performance, you must set the Sysbench parameters as needed. For example, you can set the `oltp-skip-trx` **parameter to make sure that the Sysbench does not run the SQL statement in a transaction.**

Set common parameters

You can set the following parameters as needed.

Name	Description
test	Path of the test file
mysql-host	IP address of the MySQL server

Name	Description
mysql-port	Port of the MySQL server
mysql-user	User name
mysql-password	Password
mysql-db	Database for testing, which must be created in advance
oltp-tables-count	Number of created tables
oltp-table-size	Number of records generated in each table
rand-init	Whether data is randomly initialized
max-time	Stress testing duration
max-requests	Total number of requests within a stress testing duration
num-threads	Number of concurrent threads
report-interval	Reporting interval of operating logs

Set parameters for transactions and read/write SQL statements

The following parameters can affect transactions and read/write SQL statements. Therefore, you must set parameters in read/write splitting tests as needed.

Name	Description
oltp-test-mode	Indicates the test mode. This parameter is unavailable in Sysbench 0.5, so this parameter can be ignored. Possible values: • `complex`: Default value. For transactional tests. • `simple`: Simple test for read-only SQL statements. • `nontrx`: For non-transactional tests. • `sp`: Stored procedures.

Name	Description
oltp-skip-trx	Indicates whether "begin" and "commit" of SQL statements are omitted. Possible values: · `off`: Default value. All SQL statements are executed in transactions. · `on`: Non-transactional mode. If a comparative stress test is executed repeatedly, you must run `prepare` and `cleanup` again. **Note:** When a stress test is executed to test the read/write splitting performance, you must set it to `on` and omit "begin" and "commit" of SQL statements.
oltp-read-only	Indicates whether read-only SQL statements are generated. Possible values: · `off`: Default value. The mixed read/write SQL statements of `oltp.lua` is executed. · `on`: Only read-only SQL statements are generated. UPDATE, DELETE, and INSERT SQL statements are not applicable. **Note:** Set the parameter value as needed to perform read-only or read/write tests.

Stress testing examples

Test read/write performance

1. **Run the following command to prepare data:**

```
sysbench --test=./tests/db/oltp.lua --mysql-host=127.0.0.1 --mysql
-port=3001 --mysql-user=abc --mysql-password=abc123456 --mysql-db
=testdb --oltp-tables-count=10 --oltp-table-size=500000 --report-
```

```
interval=5 --oltp-skip-trx=on --oltp-read-only=off --rand-init=on --
max-requests=0 --max-time=300 --num-threads=100 prepare;
```

2. **Run the following command to conduct the test:**

> **Note:**
> **When data is updated for non-transactional read/write tests, errors such as `ALERT: Error 1062 Duplicate entry 'xxx' for key 'PRIMARY'` may occur. You must add `--mysql-ignore-errors=1062` to skip these errors. If the parameter `mysql-ignore-errors` does not take effect, your current Sysbench version is too old and you must upgrade it to the latest version.**

```
sysbench --test=./tests/db/oltp.lua --mysql-host=127.0.0.1 --mysql
-port=3001 --mysql-user=abc --mysql-password=abc123456 --mysql-db
=testdb --oltp-tables-count=10 --oltp-table-size=500000 --report-
interval=5 --oltp-skip-trx=on --oltp-read-only=off --mysql-ignore
-errors=1062 --rand-init=on --max-requests=0 --max-time=300 --num-
threads=100 run;
```

3. **Run the following command to clean up data:**

```
sysbench --test=./tests/db/oltp.lua --mysql-host=127.0.0.1 --mysql
-port=3001 --mysql-user=abc --mysql-password=abc123456 --mysql-db
=testdb --oltp-tables-count=10 --oltp-table-size=500000 --report-
interval=5 --oltp-skip-trx=on --oltp-read-only=off --rand-init=on --
max-requests=0 --max-time=300 --num-threads=100 cleanup;
```

Test read-only performance

1. **Run the following command to prepare data:**

```
sysbench --test=./tests/db/oltp.lua --mysql-host=127.0.0.1--mysql
-port=3001--mysql-user=abc --mysql-password=abc123456 --mysql-db
=testdb --oltp-tables-count=10--oltp-table-size=500000--report-
interval=5--oltp-skip-trx=on --oltp-read-only=on --rand-init=on --
max-requests=0--max-time=300--num-threads=100 prepare;
```

2. **Run the following command to conduct the test:**

```
sysbench --test=./tests/db/oltp.lua --mysql-host=127.0.0.1 --mysql
-port=3001 --mysql-user=abc --mysql-password=abc123456 --mysql-db
=testdb --oltp-tables-count=10 --oltp-table-size=500000 --report-
interval=5 --oltp-skip-trx=on --oltp-read-only=on --rand-init=on --
max-requests=0 --max-time=300 --num-threads=100 run;
```

3. **Run the following command to clean up data:**

```
sysbench --test=./tests/db/oltp.lua --mysql-host=127.0.0.1 --mysql
-port=3001 --mysql-user=abc --mysql-password=abc123456 --mysql-db
=testdb --oltp-tables-count=10 --oltp-table-size=500000 --report-
```

```
interval=5 --oltp-skip-trx=on --oltp-read-only=on --rand-init=on --
max-requests=0 --max-time=300 --num-threads=100 cleanup;
```

7.11 Verify the effect of read/write splitting

7.11.1 Use SQL Explorer to verify the effect of read/Write splitting

This topic describes how to use SQL Explorer to verify the effect of read/write splitting by comparing the number of SQL statements executed by each of the master and read-only instances.

For information about how to enable SQL Explorer, see *SQL Explorer*.

7.11.2 Use internal SQL statements to verify the effect of read/write splitting

This topic describes how to verify the effect of read/write splitting for an RDS for MySQL instance by running the `/*PROXY_INTERNAL*/show last route;` command.

Note:

Currently, do not use this SQL statement in the production environment because it is still being tested internally and may be adjusted later as required.

View the database to which an SQL statement is sent for execution

You can view the ID of the instance on which an SQL statement is executed by running the following command:

```
/*PROXY_INTERNAL*/show last route;
```

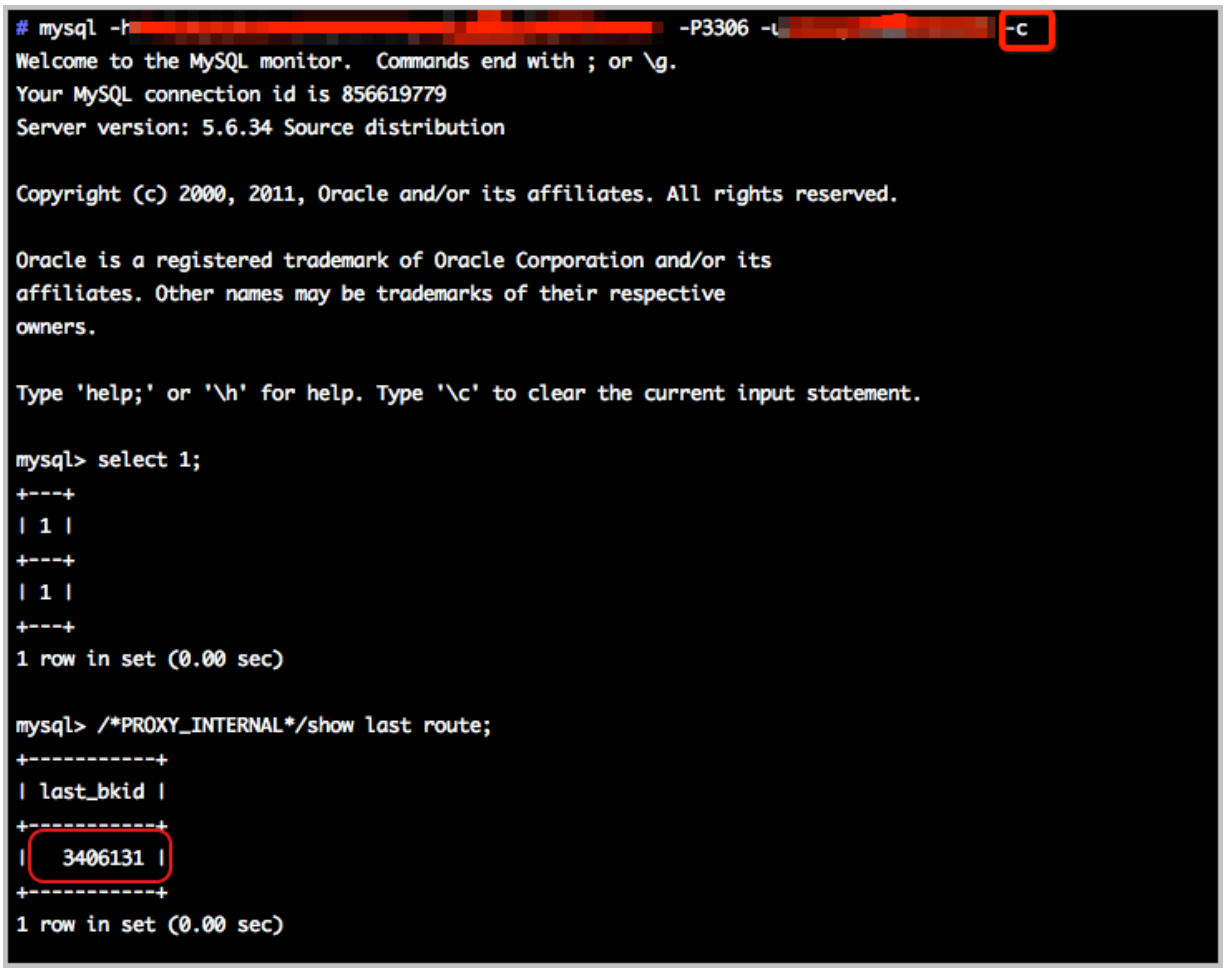
Note:

RDS provides a built-in hint SQL statement (which can only be executed by using read/write splitting VIP). When running this statement on the MySQL client, the `-c` option must be selected. Otherwise, the client filters out the hit and the following error is returned:

```
ERROR 1064 (42000): You have an error in your SQL syntax; check the
manual that
corresponds to your MySQL server version for the right syntax to use
near 'last route' at
```

```
line 1
```

The returned result `last_bkid` indicates the ID of the database to which the last SQL statement (the one before the hit) was sent. This ID is the unique identification of an RDS instance. The following figure shows the details.



Note:

Given that the SQL load for RDS is measured in batches with a minimum unit of 100 entries, the first 100 `select` statements are executed on the same instance before the 101st `select` statement is routed to another instance. To verify this, you can write a simple SQL file such as *`a.sql`*:

```
select 1;
/*PROXY_INTERNAL*/show last route;select 1;
***100 entries***;
select 1;
```

```
/*PROXY_INTERNAL*/show last route;
```

Now you can see that the 101st SQL statement is routed to another instance (assuming that more than two read-only instances are available for sharing the load).

Verify that all write requests are forwarded to the master database (master instance) for execution

Once read/write splitting is enabled for RDS instances, write requests can only be forwarded to the master database because read-only databases only process read requests. Even when a system or routing error occurs, for example, a write SQL statement is routed to a read-only database, such write request can be routed back to the master database for execution according to the error reason (read_only error).

Additionally, you can run the `insert` statement, and then run the following hint SQL statement to verify that all write requests are forwarded to the master database.

```
/*PROXY_INTERNAL*/show last route;
```

7.12 Verify the allocated read weights

This topic describes how to verify the allocated read weights when read/write splitting is enabled.

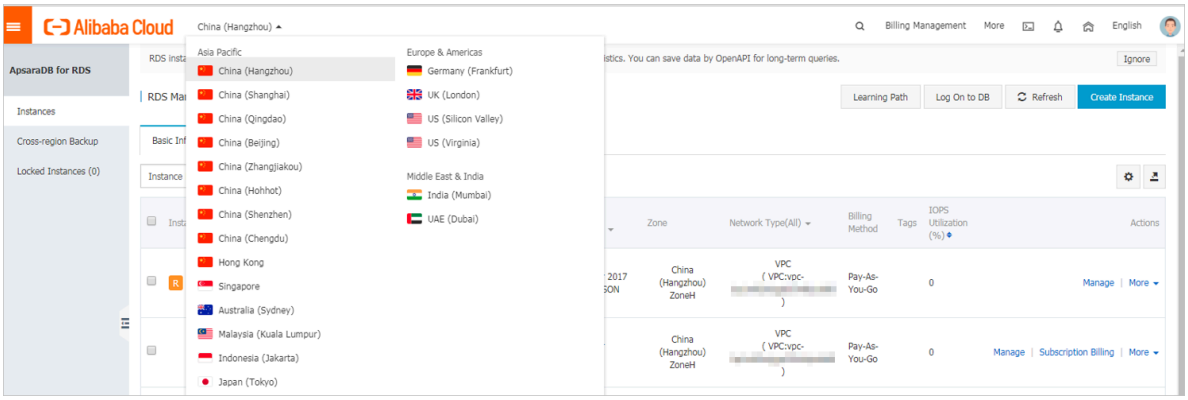
To verify the load ratio of each read weight, you can run the `select @@server_id;` command for 10,000 times by using persistent connections and collect the number of times that each `server_id` appears in the output.

Alternatively, you can check whether the load ratio of read weight is consistent with the distributed ratio by using one of the following two methods:

Verify the load ratio based on monitoring data in the console

1. **Log on to the *RDS console*.**

2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Monitoring and Alerts.
5. On the Monitoring tab, select the monitoring type Engine Monitoring.
6. View the TPS (Transactions per Second)/QPS (Queries per Second) chart to obtain the number of read and write operations on each of the master and read-only instances.

> **Note:**
> Refreshing the TPS/QPS chart takes about five minutes.

7. Compare the QPS/TPS of each instance to verify whether the load ratio is correct.

Verify the SQL load by directly connecting to each instance

You can view the number of SQL statements executed by each instance by connecting to the master database and read-only databases involving read/write splitting.

> **Note:**
> To verify this, the connection addresses of the master database and read-only databases instead of the read/write splitting address are needed.

Run either of the following commands to verify the SQL load:

```
select * from information_schema.global_status where VARIABLE_NAME = '
COM_SELECT';
```

```
select * from information_schema.global_status where VARIABLE_NAME = '
COM_INSERT;
```

7.13 Rules of weight allocation by the system

This topic describes the default read weights allocated to RDS instances of various specifications.

Weight values list

When the read weights are automatically allocated for instances by the system, the values of these weights are fixed, as shown in the following table.

Table 7-2: Weights for RDS for MySQL read-only instances

Specification code	Specification type	Memory	CPU	Weight
rds.mysql.t1.small	General-purpose instance	1 GB	1	100
rds.mysql.s1.small	General-purpose instance	2 GB	1	100
rds.mysql.s2.large	General-purpose instance	4 GB	2	200
rds.mysql.s2.xlarge	General-purpose instance	8 GB	2	200
rds.mysql.s3.large	General-purpose instance	8 GB	4	400
rds.mysql.m1.medium	General-purpose instance	16 GB	4	400

Specification code	Specification type	Memory	CPU	Weight
rds.mysql.c1.large	General-purpose instance	16 GB	8	800
rds.mysql.c1.xlarge	General-purpose instance	32 GB	8	800
rds.mysql.c2.xlarge	General-purpose instance	64 GB	16	1600
rds.mysql.c2.xlp2	General-purpose instance	96 GB	16	1600
rds.mysql.c2.2xlarge	General-purpose instance	128 GB	16	1600
mysqlro.x8.medium.1	Dedicated instance	16 GB	2	200
mysqlro.x4.large.1	Dedicated instance	16 GB	4	200
mysqlro.x8.large.1	Dedicated instance	32 GB	4	400
mysqlro.x4.xlarge.1	Dedicated instance	32 GB	8	400
mysqlro.x8.xlarge.1	Dedicated instance	64 GB	8	800
mysqlro.x4.2xlarge.1	Dedicated instance	64 GB	16	800
mysqlro.x8.2xlarge.1	Dedicated instance	128 GB	16	1600
mysqlro.x4.4xlarge.1	Dedicated instance	128 GB	32	1600
rds.mysql.st.d13	Dedicated-host instance	220 GB	30	3000
rds.mysql.st.h43	Dedicated-host instance	470 GB	60	6000

Specify whether a SQL statement is sent to the master instance or a read-only instance by adding a hint

In addition to the weight distribution system of read/write splitting, hints serve as a complementary SQL syntax to specify whether a SQL statement is executed on the master instance or a read-only instance.

Hints supported by RDS read/write splitting are as follows:

- `/*FORCE_MASTER*/`**: specifies that a SQL statement is executed on the master instance.**
- `/*FORCE_SLAVE*/`**: specifies that a SQL statement is executed on a read-only instance.**

> **Note:**
> **When you use the `/*FORCE_SLAVE*/` hint in a SQL statement, the SQL statement is routed to the master instance if the read weight of the master instance is not 0.**

For example, after a hint is prefixed to the following statement, the statement is always routed to and executed on the master instance regardless of the preset weight:

```
/*FORCE_MASTER*/ SELECT * FROM table_name;
```

7.14 FAQ for read/write splitting

This topic describes the FAQ related to read/write splitting.

Why does the new read weights not take effect?

After you change the read weights, RDS allocates read and write requests based on the new weights only after a new connection is established. However, your application does not terminate the existing connection or reestablish a connection.

Why does the load on each of the master and read-only instances not comply with the specified read weight?

If the load on each of the master and read-only instances does not comply with the specified read weight, check the following two aspects:

- **Whether the request statements contain transactions. All requests that contain transactions, including transaction read requests, are routed only to the master instance.**
- **Whether your application connects to each of the master and read-only instances only by using the read/write splitting address. If your application connects to any master or read-only instance by using the endpoint of this specific instance, RDS does not allocate requests to this instance based on the specified weight.**

Why does RDS route read requests to the master instance when the read weight of the master master is 0?

Check the following two aspects:

- **Whether the request statements contain transactions. All requests that contain transactions, including transaction read requests, are routed only to the master instance.**
- **Whether the unavailability duration or replication latency of each read-only instance with a specified read weight exceeds the preset threshold. If so, RDS determines that all the read-only instances are unavailable and therefore routes read requests to the master instance.**

How does read/write splitting guarantee the timeliness of data reading?

For more information, see *How does read/write splitting ensure the timeliness of data reading?***.**

How do I set read weights by calling API actions?

For more information, see *#unique_121***.**

How do I select the network type of the read/write splitting address?

If your application is deployed in the same VPC as the RDS instance, we recommend that you use the internal endpoint to guarantee data security and communication efficiency. If your application is deployed on your computer or in a different network from the RDS instance, you can use the public endpoint to establish a connection. For more information about how to switch between network types, see *Change the network type of the read/write splitting address for an RDS for MySQL instance***.**

What is the difference between the read/write splitting address and the internal and public endpoints of the master instance?

After read/write splitting is enabled, a read/write splitting address is generated. You must add the read/write splitting address and the read weight of each instance

to the configuration data in your application. Then, after the requests from the application reach this address, RDS automatically allocates them to each of the master and read-only instances based on the request types and the read weights.

If the connection address in your application is the internal or public endpoint of the master instance, then all the requests are routed to the master instance.

Can I change the read/write splitting address?

No. You cannot change the read/write splitting address. The read/write splitting address is fixed. It does not change as you enable or disable read/write splitting , therefore you do not need to modify the configuration data in your application frequently. This reduces maintenance costs.

Does adding a read-only instance have any impact on the master instance?

No. Adding a read-only instance does not have any impact on the master instance because the read-only instance replicates data from the slave instance.

8 Account management

8.1 Create an account for an RDS for MySQL instance

This topic describes how to create an account for an RDS for MySQL instance.

Account types

RDS for MySQL supports two types of database accounts: premier accounts and standard accounts. You can manage all your accounts and databases in the console. For specific permissions, see *Account permissions*.

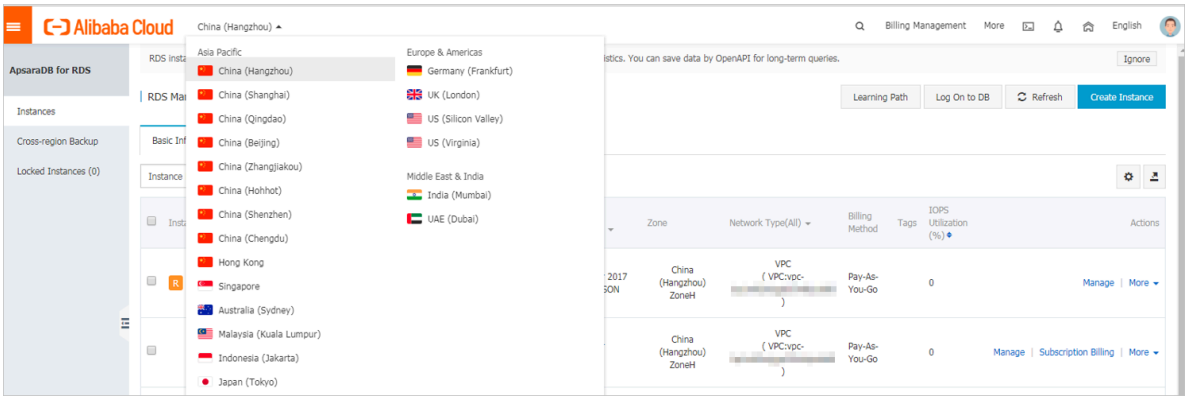
Account type	Description
Premier account	· Can only be created and managed through the console or API. · Each instance can have only one premier account, which can be used to manage all databases and standard accounts. · Has more permissions than standard accounts and can manage permissions at a more fine-grained level. For example, it can assign table-level query permissions to other accounts. · Can disconnect the connections established by any other accounts.
Standard account	· Can be created and managed through the console, API, or SQL statements. · Each instance can have up to 200 standard accounts. · Need to be manually granted with database permissions. · Cannot create or manage other accounts, or terminate the connections established by other accounts.

Account Type	Number of databases	Number of tables	Number of users
Premier account	Unlimited	< 200,000	Varies depending on the instance kernel parameters.
Standard account	500	< 200,000	Varies depending on the instance kernel parameters.

Create a premier account

1. **Log on to the *RDS console*.**

2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click its ID.
4. In the left-side navigation pane, click Accounts.
5. Click Create Account.
6. Set the following parameters.

Parameter	Description
Database Account	The account name contains 2 to 16 characters, including lowercase letters, digits, and underscores (_). It must begin with a letter and end with a letter or digit.
Account Type	Select Premier Account.
Password	The password contains 8 to 32 characters, including at least three of the following types of characters: uppercase letters, lowercase letters, digits, and special characters. The allowed special characters are as follows: ! @ # $ % ^ & * ( ) _ + - =
Re-enter Password	Enter the password again.
Note	Optional. Enter the other account information that helps to better manage the account. You can enter up to 256 characters.

7. Click OK.

Reset the permissions of a premier account

If the premier account of an RDS instance is abnormal (for example, the account permissions are unexpectedly revoked), you can reset the permissions.

1. Log on to the *RDS console*.
2. In the upper-left corner, select the region where the target RDS instance is located.

3. Find the target RDS instance and click its ID.
4. In the left-side navigation pane, click Accounts.
5. Find the premier account and in the Actions column click Reset Permissions.
6. Enter the password of the premier account and click OK.

Create a standard account

1. Log on to the *RDS console*.
2. In the upper-left cornere, select the region where the target RDS instance is located.

3. Find the target RDS instance and click its ID.
4. In the left-side navigation pane, click Accounts.
5. Click Create Account.

6. Set the following parameters.

Parameter	Description
Database Account	The account name contains 2 to 16 characters, including lowercase letters, digits, or underscores (_). It must begin with a letter and end with a letter or digit.
Account Type	Select Standard Account.
Authorized Databases	Grant permissions on one or more databases to the account. This parameter is optional. You can choose to grant permissions to the account after the account is created. a. Select one or more databases from the left area and click Authorize > to add them to the right area. b. In the right area, click Read/Write, Read-only, DDL Only, or DML Only. If you want to grant the permissions for multiple databases in batches, select all the databases and in the upper-right corner click the button such as Full Control Read/Write. Note: The button in the upper-right corner changes as you click. For example, after you click Full Control Read/Write, the permission changes to Full Control Read-only.
Password	The password must contain 8 to 32 characters, including at least three of the following types of characters: uppercase letters, lowercase letters, digits, and special characters. The allowed special characters are as follows: ! @ # $ % ^ & * ( ) _ + - =
Re-enter Password	Enter the password again.
Note	Optional. Enter the other account information that helps to better manage the account. You can enter up to 256 characters.

7. Click OK.

Account permissions

Account type	Permiss type	Permission				
Premier account	N/A	SELECT	INSERT	UPDATE	DELETE	CREATE
		DROP	RELOAD	PROCESS	REFERENCES	INDEX
		ALTER	CREATE TEMPORARY TABLES	LOCK TABLES	EXECUTE	REPLICATION SLAVE
		REPLICATION CLIENT	CREATE VIEW	SHOW VIEW	CREATE ROUTINE	ALTER ROUTINE
		CREATE USER	EVENT	TRIGGER	N/A	N/A
Standard account	Read-only	SELECT	LOCK TABLES	SHOW VIEW	PROCESS	REPLICATION SLAVE
		REPLICATION CLIENT	N/A	N/A	N/A	N/A
	Read/ write	SELECT	INSERT	UPDATE	DELETE	CREATE
		DROP	REFERENCES	INDEX	ALTER	CREATE TEMPORARY TABLES
		LOCK TABLES	EXECUTE	CREATE VIEW	SHOW VIEW	CREATE ROUTINE
		ALTER ROUTINE	EVENT	TRIGGER	PROCESS	REPLICATION SLAVE
		REPLICATION CLIENT	N/A	N/A	N/A	N/A
	DDL only	CREATE	DROP	INDEX	ALTER	CREATE TEMPORARY TABLES
		LOCK TABLES	CREATE VIEW	SHOW VIEW	CREATE ROUTINE	ALTER ROUTINE
		PROCESS	REPLICATION SLAVE	REPLICATION CLIENT	N/A	N/A
	DML only	SELECT	INSERT	UPDATE	DELETE	CREATE TEMPORARY TABLES

Account type	Permiss type	Permission				
		LOCK TABLES	EXECUTE	SHOW VIEW	EVENT	TRIGGER
		PROCESS	REPLICATION SLAVE	REPLICATION CLIENT	N/A	N/A

APIs

API	Description
#unique_38	Used to create an account for an RDS instance.

8.2 Reset the password of an account for an RDS for MySQL instance

This topic describes how to reset the password of an account for an RDS for MySQL instance in case that the password is lost.

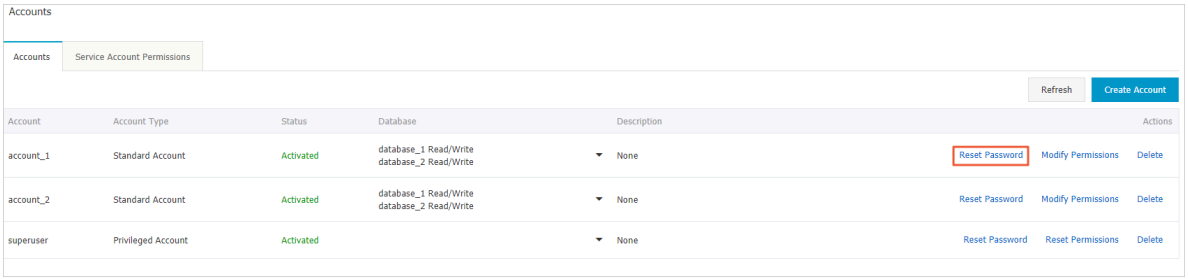
Note:
For data security purposes, we recommend you change the password on a regular basis.

Procedure

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**

3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Accounts.**

5. On the Accounts tab, select the account whose password you want to reset, and in the Actions column click Reset Password.



6. In the Reset Account Password dialog box, enter a new password and confirm it, then click OK. The password consists of 6 to 32 characters including letters, digits, hyphen (-), or underscores (_). A previously used password is not recommended.

APIs

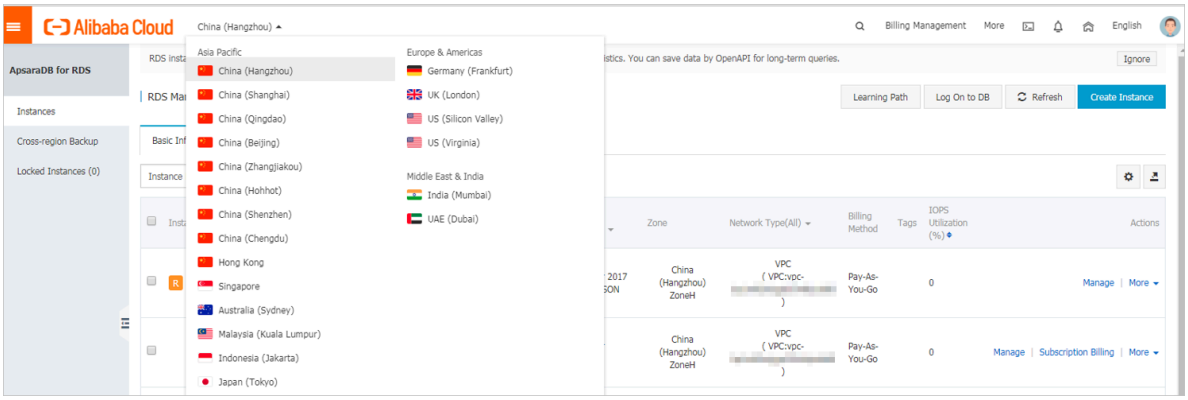
API	Description
#unique_131	Used to reset the password of a database account.

8.3 Change the permissions of an account for an RDS for MySQL instance

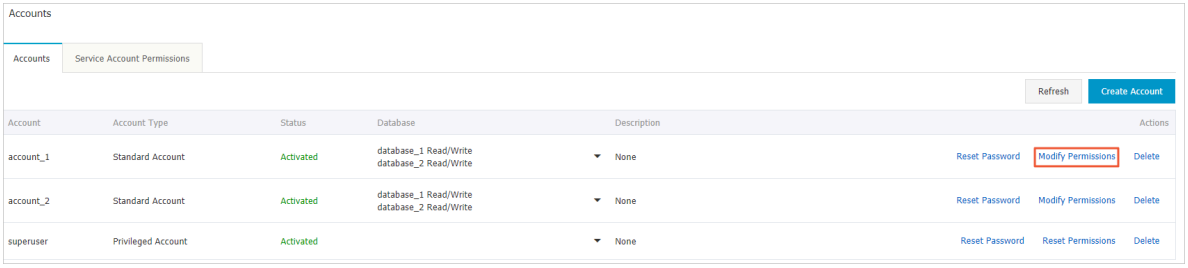
This topic describes how to change the permissions of a standard account for an RDS for MySQL instance. The permissions of the premier account cannot be changed. You can only reset the premier account if needed.

Procedure

1. Log on to the *RDS console*.
2. In the upper-left corner, select the region where the target RDS instance is located.



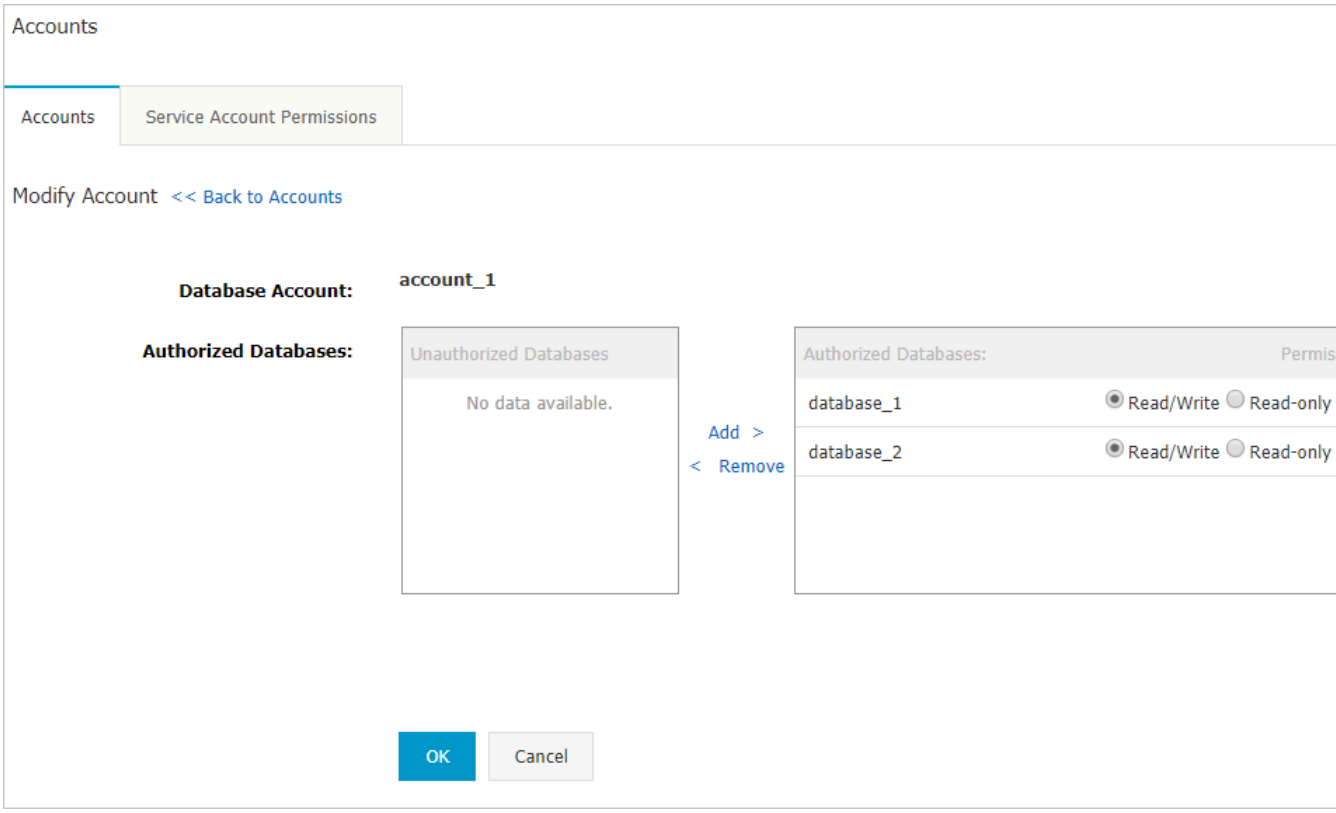
3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Accounts.
5. On the Accounts tab, find the target account and in the Actions column click Modify Permissions.



6. Change the account permissions and click OK.
   - Add an authorized database: Select a database in the Unauthorized Databases section and then click Add > to add it to the Authorized Databases section.
   - Delete an authorized database: Select a database in the Authorized Databases section and then click < Remove to add it to the Unauthorized Databases section.
   - Change the permissions of an authorized database: Find the database in the Authorized Databases section and select Read/Write, Read-only, DDL Only, or DML Only. To change the permissions of more than one authorized database in batches, you can click Full Control Read/Write, Full Control Ready-only, Full Control DDL Only, or Full Control DML Only in the upper-right corner of the Authorized Databases section.

Note:

Either of them is displayed at a time.

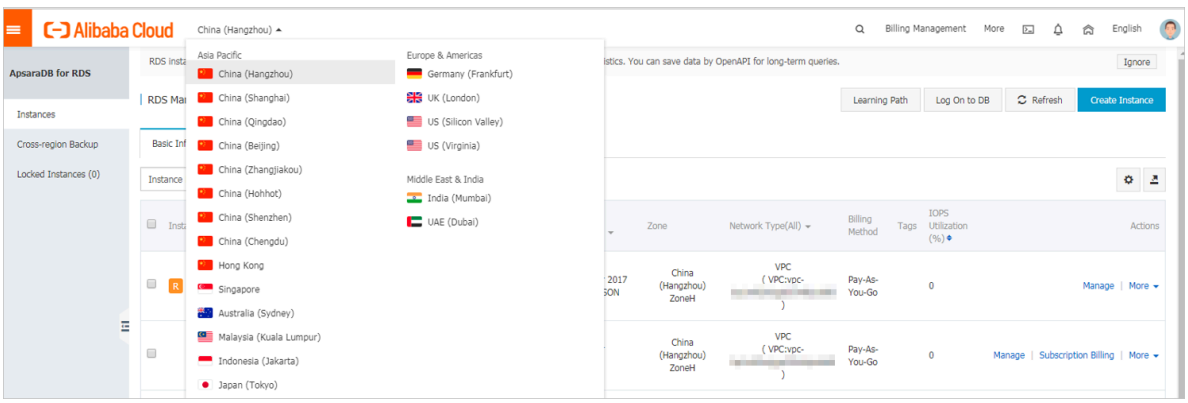


8.4 Reset the permissions of a premier account

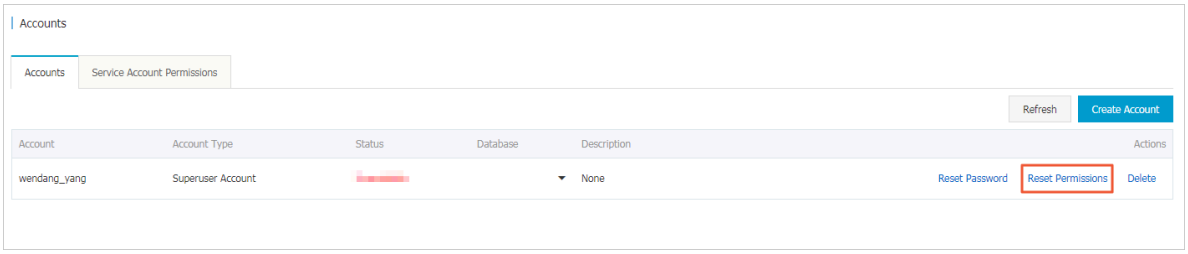
This topic describes how to reset the permissions of a premier account when the premier account becomes abnormal, for example, when the permissions of the premier account are revoked unexpectedly.

Procedure

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Accounts.
5. On the Accounts tab, find the premier account and in the Actions column click Reset Permissions.



6. Enter the password of the premier account and click OK.

8.5 Authorize a service account for an RDS for MySQL instance

This topic describes how to authorize a service account for an RDS for MySQL instance.

If you are seeking for technical support from Alibaba Cloud and if it is necessary to operate your DB instance during technical support, you must authorize a service account that is used by the technical support staff to provide technical support services.

Note:

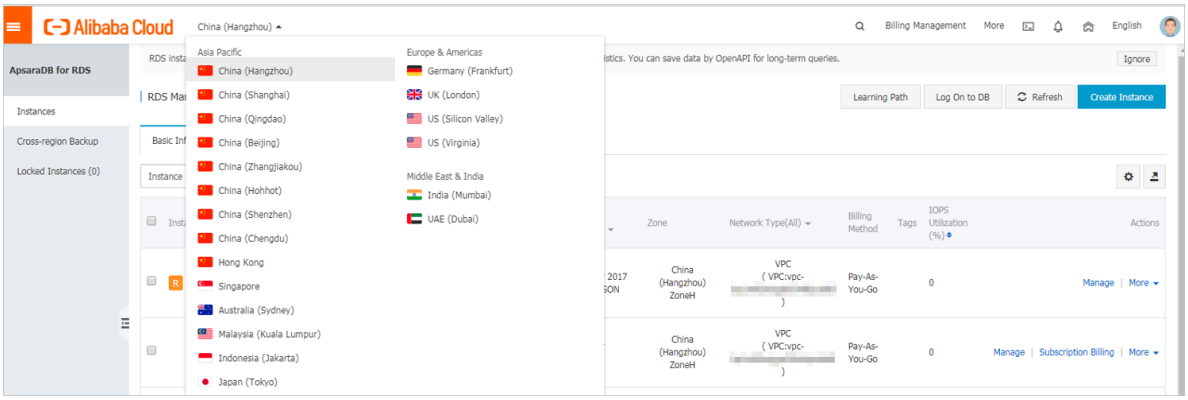
This function is available only to the following DB engine versions and editions:

- MySQL 8.0 High-availability Edition (with local SSDs)
- MySQL 5.7 Enterprise Edition
- MySQL 5.7 High-availability Edition (with local SSDs)
- MySQL 5.6 High-availability Edition
- MySQL 5.5 High-availability Edition

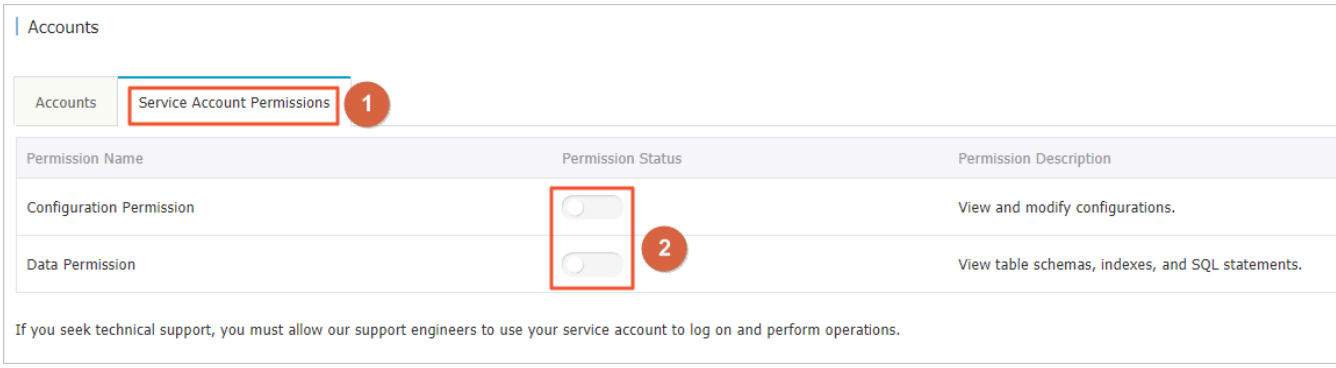
Procedure

1. Log on to the *RDS console*.

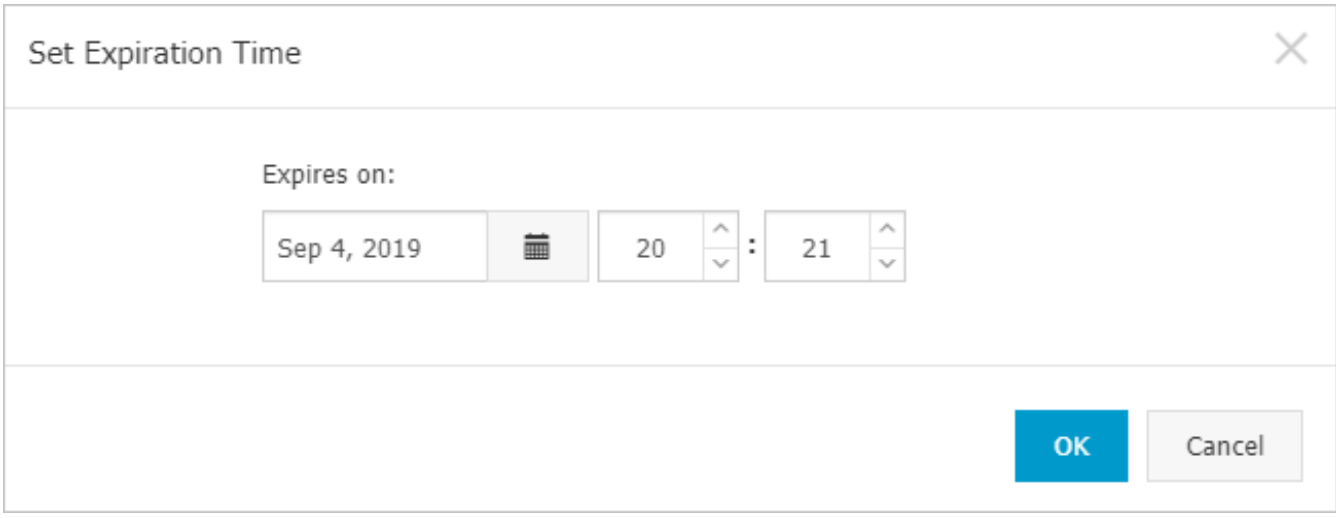
2. In the left-side navigation pane, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Accounts.
5. On the Service Account Permissions tab, select the permission to be authorized to the service account and in the Privilege Status column click the switch.
   - For troubleshooting of the IP address whitelists, database parameters, and other problems, you only need to authorize the Configuration Permission.
   - For the database performance problems caused by your application, you must authorize the Data Permission.

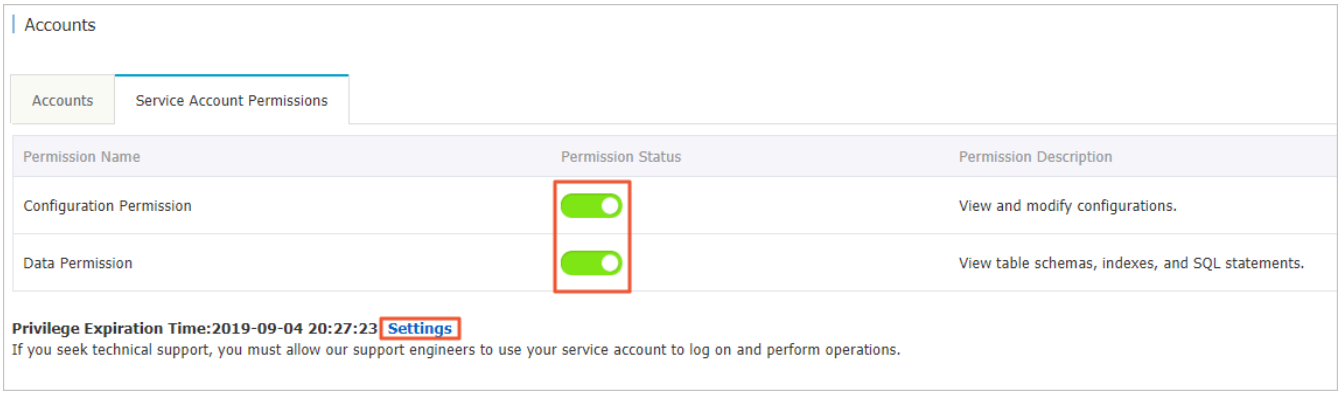


6. **In the Set Expiration Time dialog box, set the permission expiration time and click OK.**



What to do next

After you authorize permissions to a service account, you can cancel the authorization or change the authorization validity period on the Service Account Permissions tab.



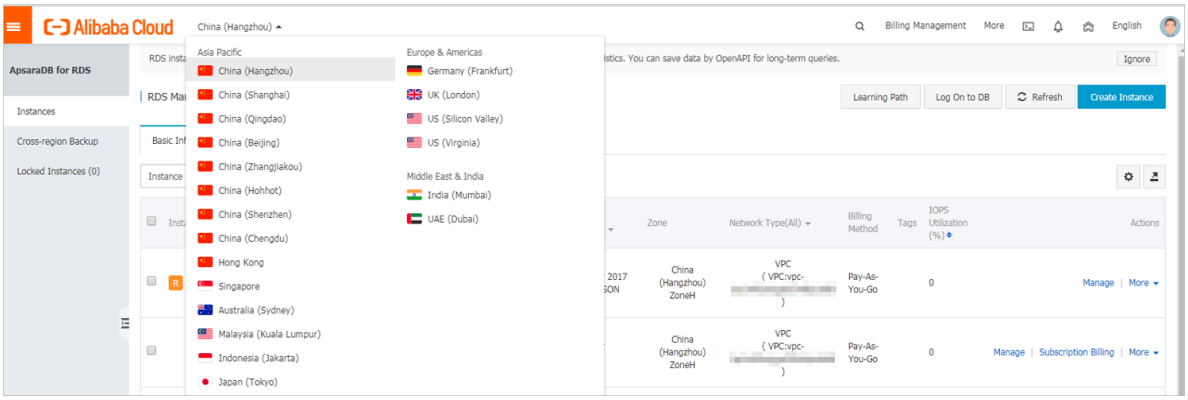
8.6 Delete an account for an RDS for MySQL instance

This topic describes how to delete an account in the RDS console or run an SQL statement to delete a standard account.

Use the console to delete an account

1. **Log on to the *RDS console*.**

2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Accounts.
5. On the Accounts tab, find the account you want to delete, and in the Actions column click Delete.
6. In the displayed dialog box, click Confirm.

Run an SQL statement to delete a standard account

You can only run an SQL statement to delete a standard account for an RDS instance .

1. Log on to the target RDS instance. For more information, see *How to connect to ApsaraDB?*
2. Run the following command to delete a standard account:

```
DROP USER 'username'@'localhost';
```

APIs

API	Description
#unique_136	Used to delete an account for an RDS instance.

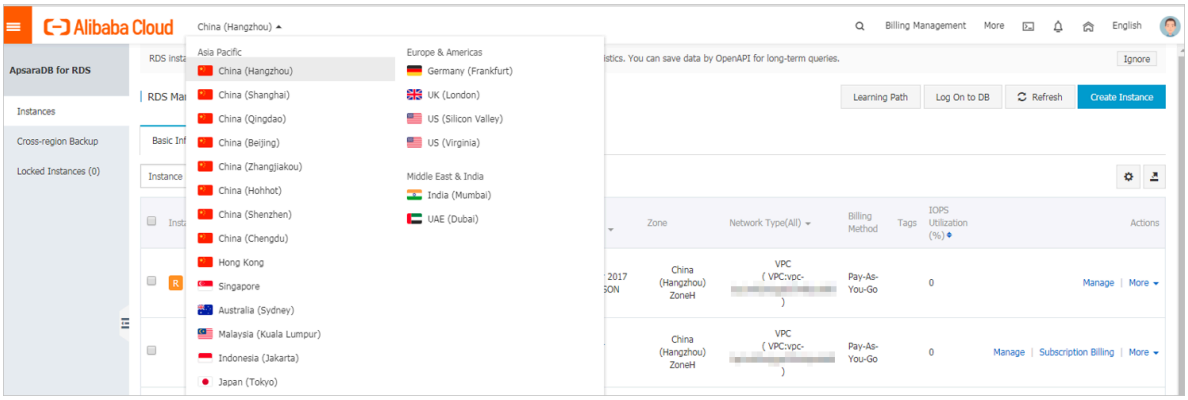
9 Database management

9.1 Create a database for an RDS for MySQL instance

This topic describes how to crate a database for an RDS for MySQL instance.

Procedure

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**



3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Databases.**
5. **Click Create Database.**
6. **Set the following parameters.**

Parameters	Description
Database Name	The account name contains 2 to 16 characters, including lowercase letters, digits, underscores (_), and hyphens (-). It must begin with a letter and end with a letter or digit. Each database name must be unique in the RDS instance.
Supported Character Set	Select utf8, gbk, latin1, or utf8mb4. If you need another character set, select all and then select the character set from the list.

Parameters	Description
Authorized Account	Select the account that needs to access this database. You can also leave this parameter blank and set the authorized account after the database is created. **Note:** Only standard accounts are displayed, because the premier account has all permissions for all databases.
Account Type	Select Read/Write, Read-only, DDL only, or DML only.
Description	Optional. Enter the other account information that helps to better manage the account. You can enter up to 256 characters.

7. Click OK.

APIs

API	Description
#unique_39	Used to create a database for an RDS instance.

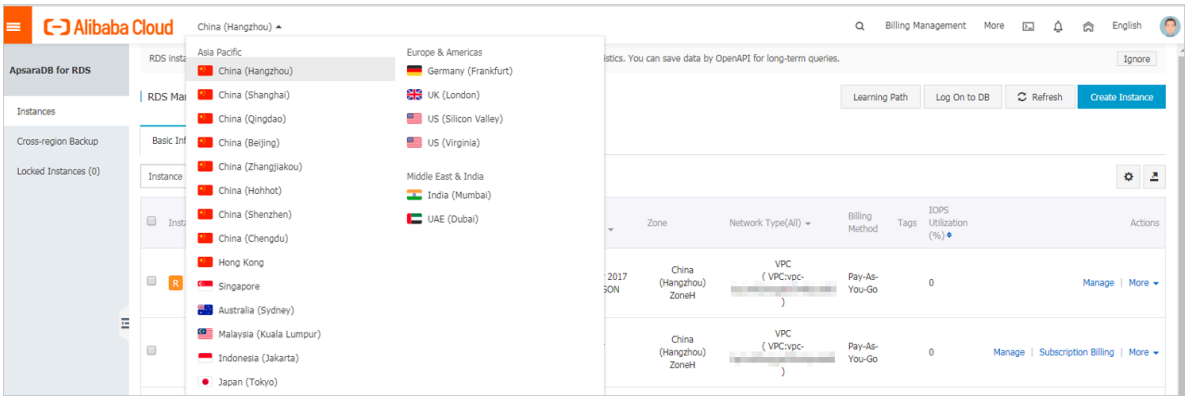
9.2 Delete a database for an RDS for MySQL instance

This topic describes how to delete a database for an RDS for MySQL instance by using the RDS console or running an SQL statement. Each method applies to different types of instances. You can choose a suitable method based on the RDS instance whose database you want to delete.

Delete a database by using the RDS console

1. Log on to the *RDS console*.

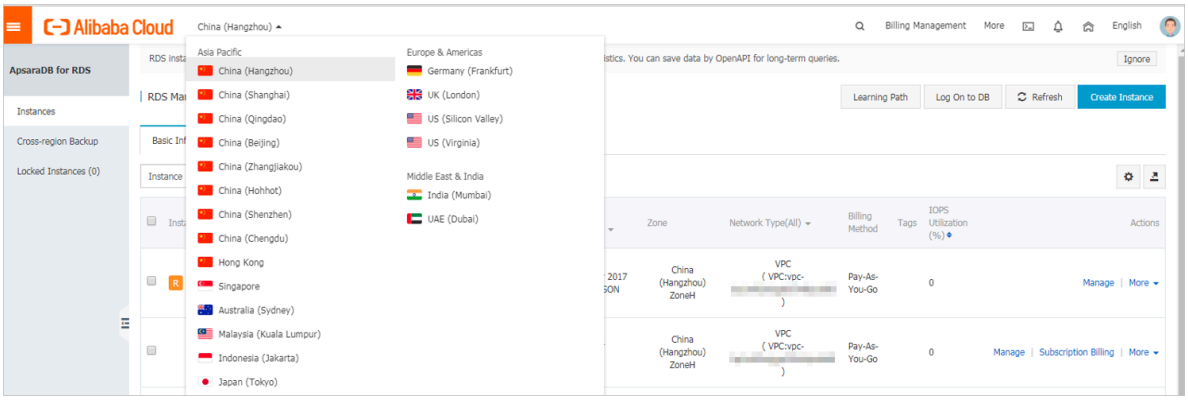
2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Databases.
5. Find the database you want to delete, and in the Actions column click Delete.
6. In the displayed dialog box, click Confirm.

Delete a database by running an SQL statement

1. Log on to the *RDS console*.
2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. In the upper-right corner, click Log On to DB go to the RDS Database Logon page of the *DMS console*.

5. On the RDS Database Logon page, check the connection address and port information. If the information is correct, enter the database username and password.

   Parameter description:

   - Databases Username: the name of the premier account.
   - Password: the password for the premier account.

   > **Note:**
   > You can view the connection address and port information of this account on the Basic Information page of the instance in the RDS console.

6. Enter the verification code and click Log On.

   > **Note:**
   > If you want the browser to remember the password, select Remember Password before you click Log On.

7. If DMS prompts you to add the IP address segment of the DMS server to an IP address whitelist of the RDS instance, click Configure Whitelist. For more information about how to manually configure the whitelist, see *Configure a whitelist for an RDS for MySQL instance*.
8. Click Log On.
9. In the top navigation bar, choose SQL Operations > SQL Window.
10. Run the following command to delete a database:

```
DROP DATABASE <database name>;
```

11. Click execute to delete the database.

APIs

API	Description
#unique_141	Used to delete a database for an RDS instance.

10 Network management

10.1 Switch to a new VPC and VSwitch for an RDS for MySQL instance

This topic describes how to switch to a new VPC and VSwitch for an RDS for MySQL instance.

Prerequisites

The DB engine version and edition are one of the following:

- **MySQL 8.0 High-availability Edition (with local SSDs)**
- **MySQL 5.7 High-availability Edition (with local SSDs)**
- **MySQL 5.6 High-availability Edition**
- **MySQL 5.5**

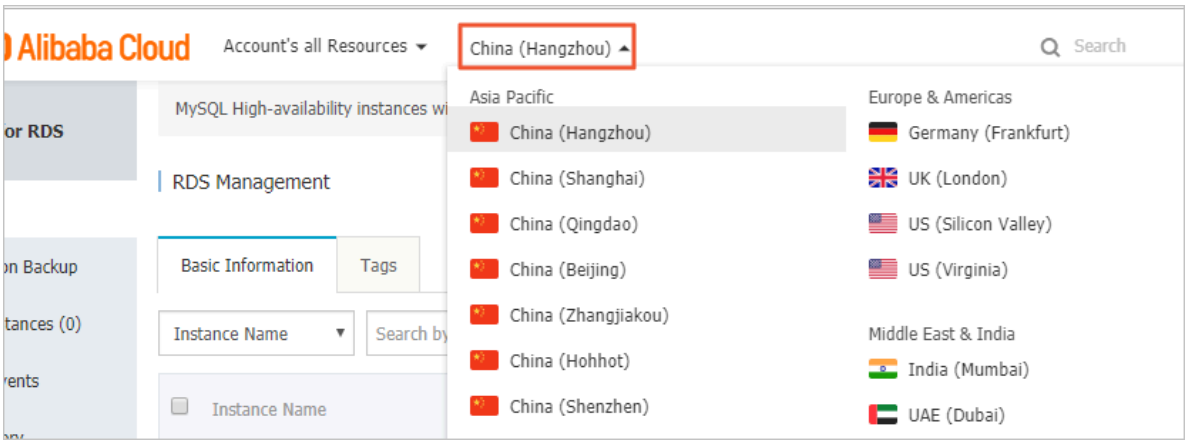
Impact

- **The RDS instance is disconnected for about 30 seconds during the switchover. Make sure that your application can automatically reconnect to the RDS instance .**
- **The virtual IP address of the RDS instance changes after the switchover. We recommend that you use *an endpoint* of the RDS instance to establish a connection.**
- **The *DRDS* is unavailable for a short period of time due to the change to the virtual IP address during the switchover. You must update and view the connection information of the RDS instance in the console immediately after the switchover is complete.**
- **The *DMS* and *DTS* is unavailable for a short period of time due to the change to the virtual IP address during the switchover, and automatically returns to normal after the switchover is complete.**
- **You must clear the cache on your database client properly. If you do not do so, data can be read but cannot be written into the RDS instance.**

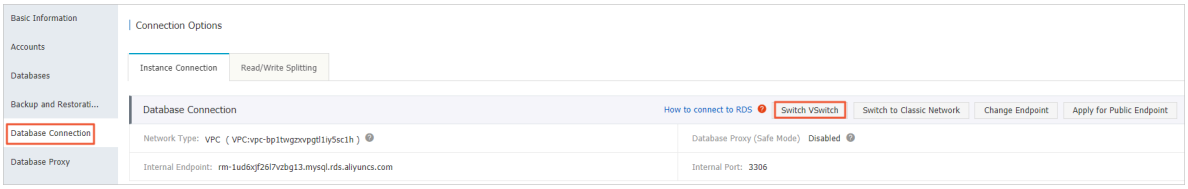
Procedure

1. **Log on to the *RDS console*.**

2. In the upper-left corner, select the region where the target RDS instance is located.



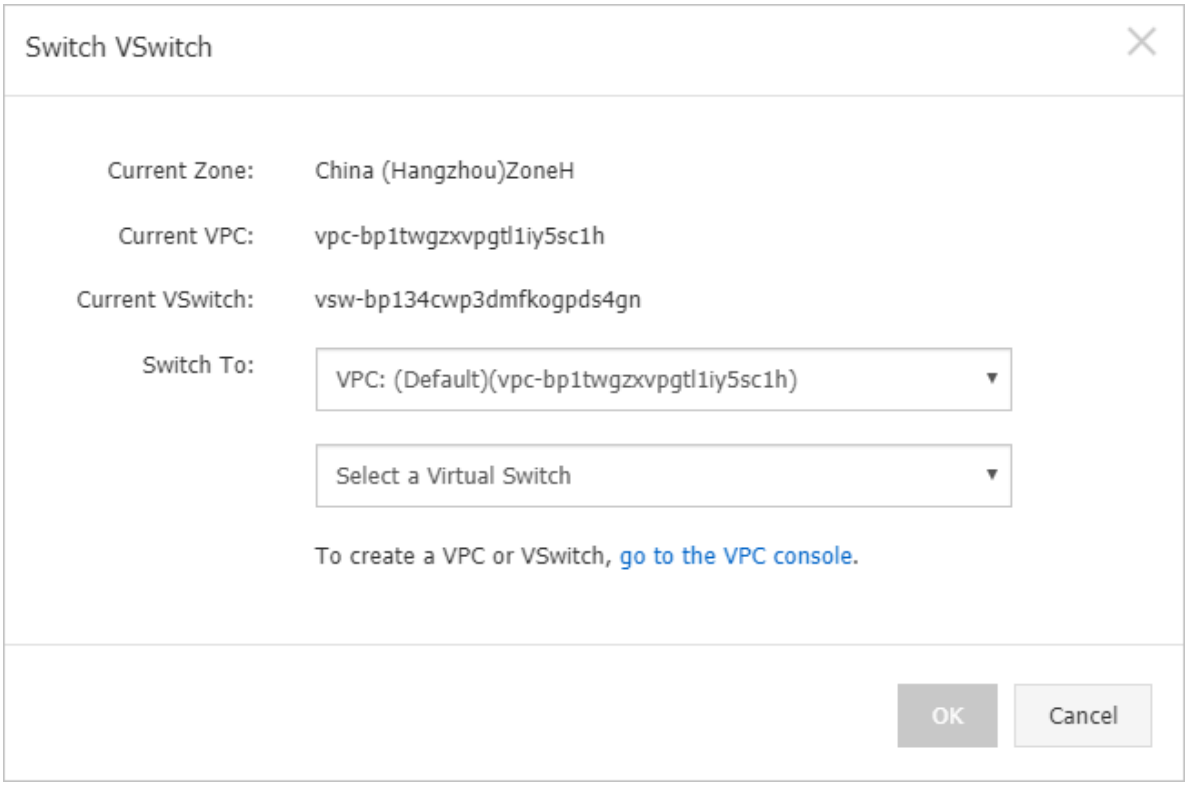
3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Database Connection.
5. In the upper-right corner, click Switch VSwitch.



6. Select a VPC and a VSwitch, then click OK.

Note:

To create a VPC or VSwitch, click go to the VPC console.



7. **In the displayed dialog box, click the button for confirming the switchover.**

10.2 Configure a hybrid access solution to smoothly migrate an RDS instance from the classic network to a VPC

This topic describes how to configure a hybrid access solution to smoothly migrate an RDS instance from the classic network to a VPC. To meet the increasing needs of migration between different network types, ApsaraDB for RDS introduces the hybrid access solution. This solution enables a smooth migration from the classic network to a VPC without any transient disconnections or service interruptions. The solution also offers the option to migrate a primary instance and its read-only instances separately without any interference with each other.

Background information

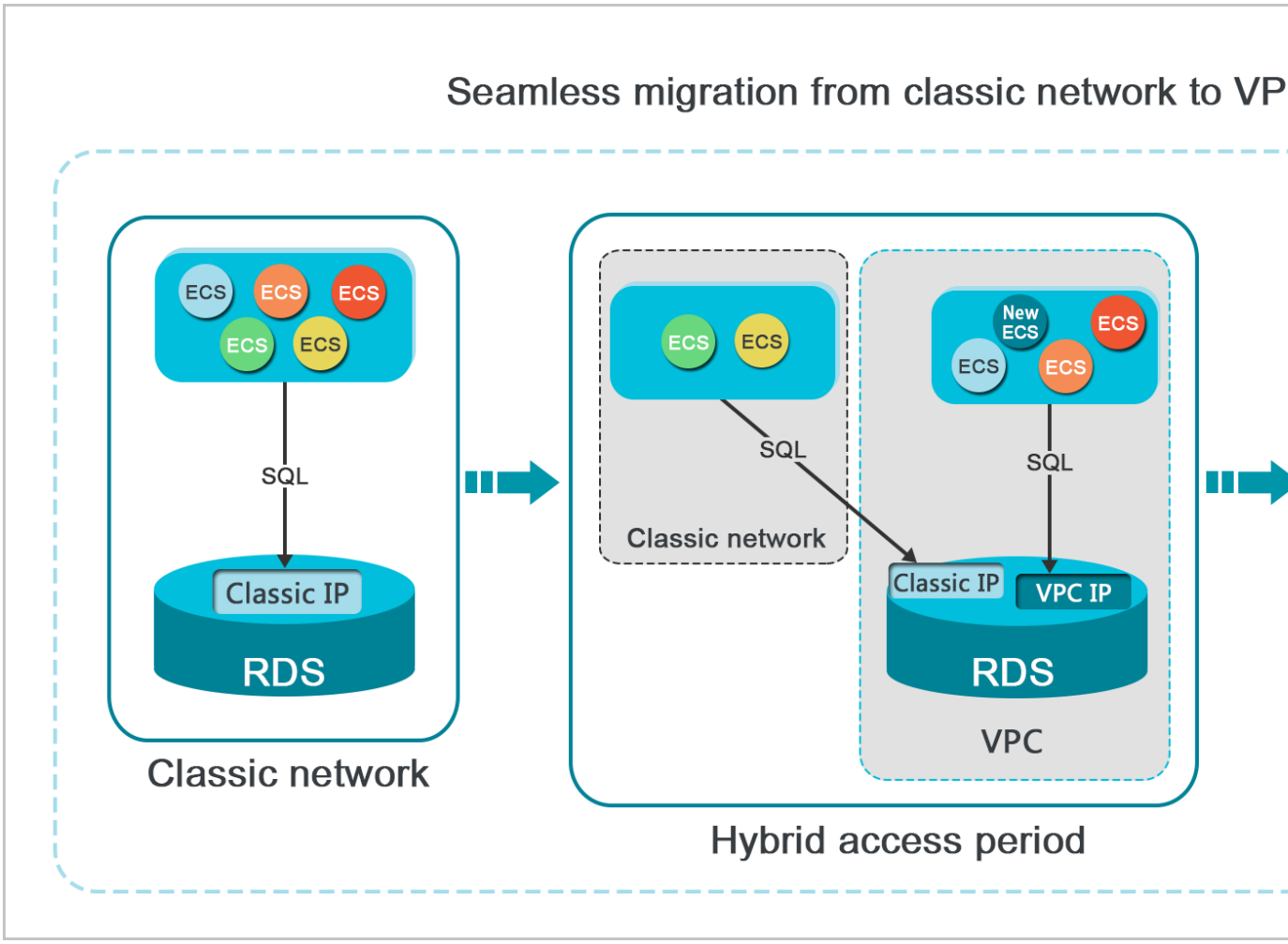
In the past, when migrating an RDS instance from the classic network to a VPC, the internal endpoint of the RDS instance changes. The connection string of the RDS instance remains the same but the IP address bound to the connection string is changed to the corresponding IP address in the VPC. This change will cause a 30-second transient disconnection, and the ECS in the classic network cannot access the RDS instance through the internal endpoint within this period. To migrate the

RDS instance across different networks in a smooth manner, ApsaraDB for RDS introduces the hybrid access solution.

Hybrid access refers to the ability of an RDS instance to be accessed by ECS on both the classic network and VPC. During the hybrid access period, the RDS instance reserves the original internal endpoint of the classic network and adds an internal endpoint of VPC. This prevents transient disconnections during the RDS database migration.

For better security and performance, we recommend that you use the internal endpoint of VPC only. Therefore, hybrid access is available for a limited period of time. The internal endpoint of the classic network is released when the hybrid access period expires. In that case, your applications cannot access the RDS database by using the internal endpoint of the classic network. You must configure the internal endpoint of VPC in all your applications during the hybrid access period. This can guarantee smooth network migration and minimize the impact on your services.

For example, your company wants to use the hybrid access solution to migrate RDS instances from the classic network to a VPC. During the hybrid access period, some applications can access the database through the internal endpoint of the VPC, and the other applications can access the database through the original internal endpoint of the classic network. When all the applications access the database through the internal endpoint of the VPC, the internal endpoint of the classic network can be released. The following figure illustrates the scenario.



Limits

During the hybrid access period, the instance has the following limits:

- **Switching to the classic network is not supported.**
- **Migrating the RDS instance to another zone is not supported.**
- **The edition of the instance cannot be switched between High-availability Edition and Enterprise Edition.**

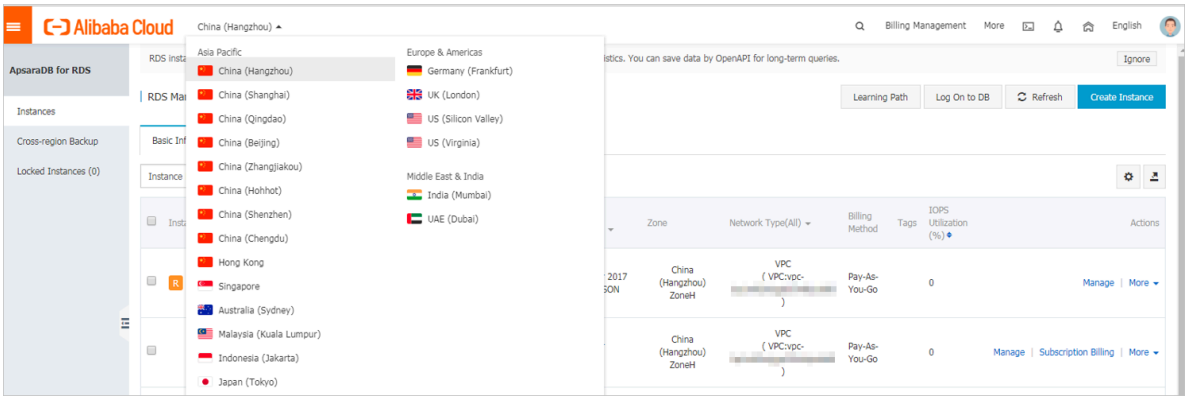
Prerequisites

- **The network type of the instance is the classic network.**
- **Available VPCs and VSwitches exist in the zone where the RDS instance is located. For more information about how to create VPCs and VSwitches, see** *Manage VPCs*.

Migrate the RDS instance from the classic network to a VPC

1. **Log on to the** *ApsaraDB for RDS console*.

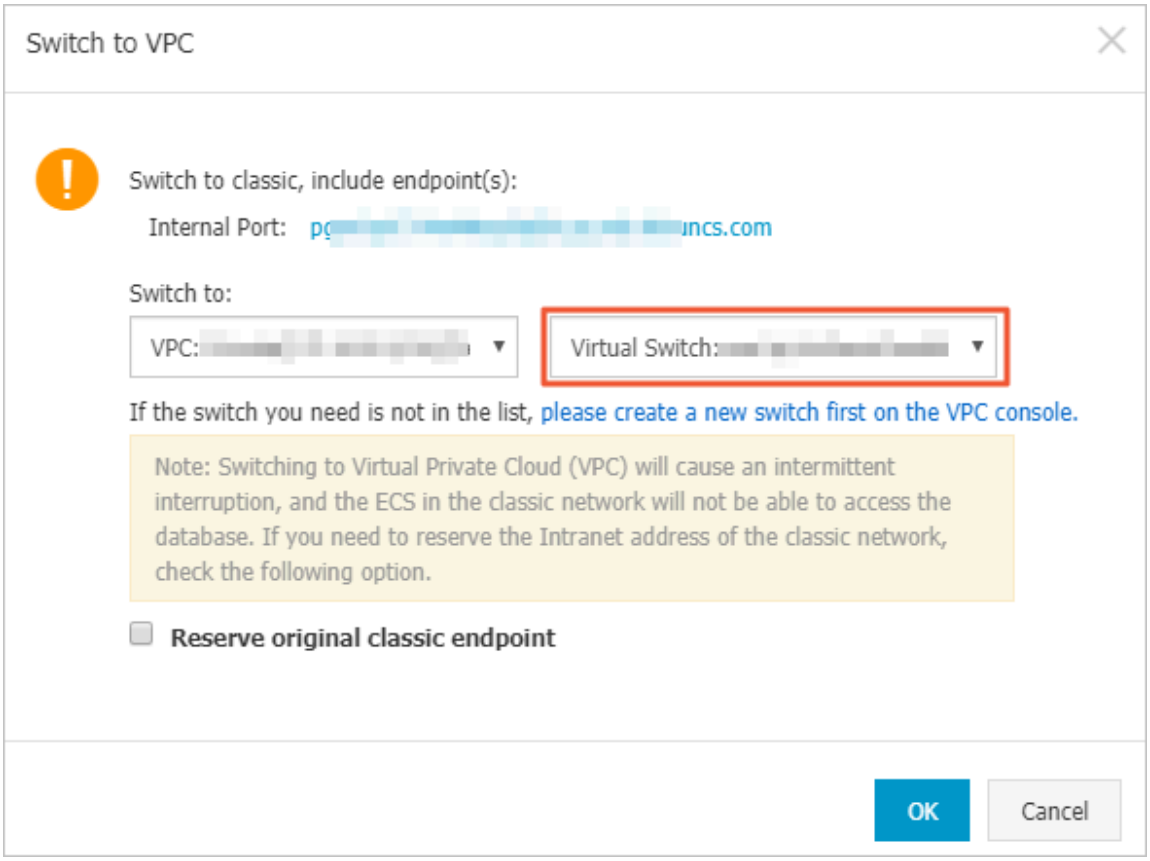
2. In the upper-left corner of the page, select the region where the instance is located.



3. Find the instance and click the instance ID.
4. In the left-side navigation pane, click Database Connections.
5. Click Switch to VPC.
6. In the dialog box that appears, select a VPC and VSwitch, and select whether to retain the internal and public endpoints of the classic network.
   - Select a VPC. We recommend that you select the VPC where your ECS instance is located. Otherwise, the ECS instance and RDS instance cannot communicate

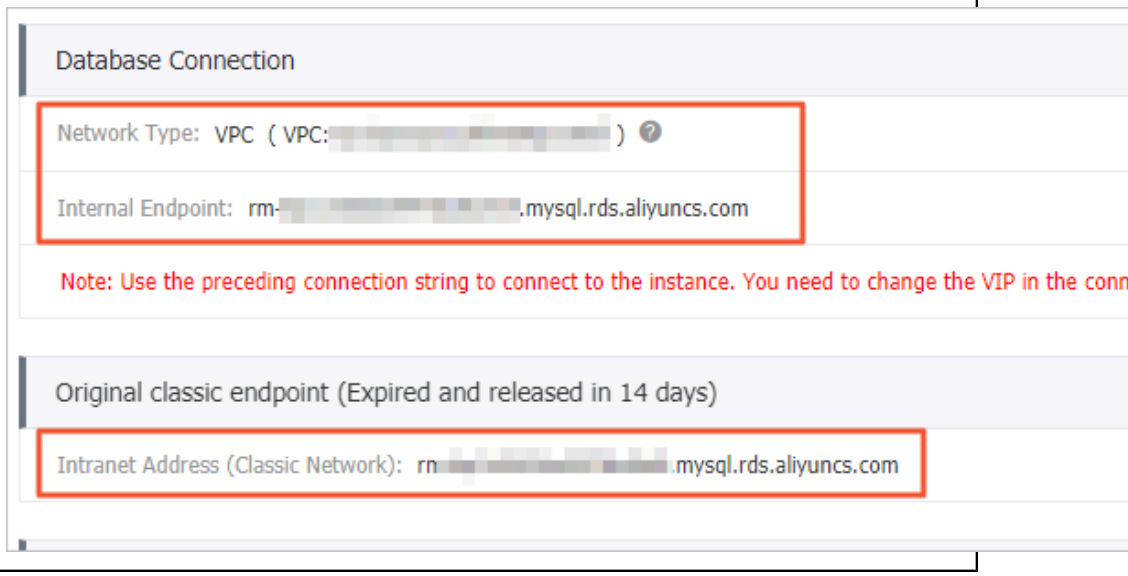
through the internal connections unless you create an express connection or gateway. For more information, see *Express connection* and *VPN gateway*.

- Select a VSwitch. If no VSwitch exists in the selected VPC (as shown in the following figure), create a VSwitch in the same zone as the instance. For more information, see *Manage VSwitches*.



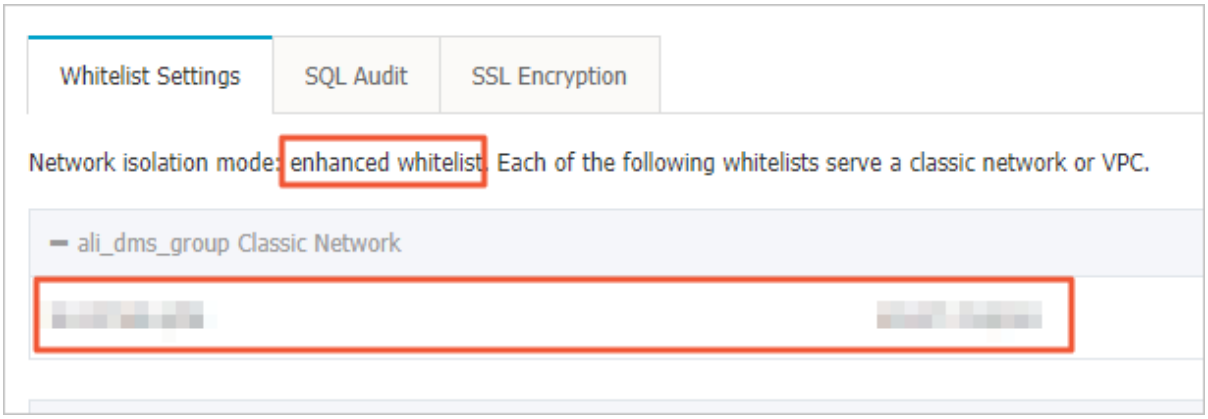
- Decide whether to select Retain Classic Network. The following table describes the different actions.

Action	Description
Clear	The endpoint of the classic network is not retained. The original endpoint is changed to the endpoint of the VPC. If the endpoint of the classic network is not retained, a 30-second transient disconnection may occur to the RDS instance when the network type is changed. The internal access to the RDS instance from the ECS instance that is located in the classic network will be immediately disconnected.

Action	Description
Select	The endpoint of the classic network is retained, and a new endpoint of the VPC is added. Indicates that the hybrid access mode is used and RDS can be simultaneously accessed by ECS instances both in the classic network and VPC through the internal endpoints. If the endpoint of the classic network is retained, the RDS instance will not be immediately disconnected when the network type is changed. The ECS instances in the classic network will not be disconnected from the internal access to the RDS instance until the internal endpoint of the classic network expires. Before the endpoint of the classic network expires, add the endpoint of the VPC to the ECS instance that is located in the same VPC. This makes sure that your business is smoothly migrated to the VPC. Within seven days before the endpoints of the classic network expire, the system will send a text message to the mobile phone bound to your account every day. 

7. Add the internal IP address of the ECS instance in the VPC to the VPC whitelist group of the RDS instance. This makes sure that the ECS instance can access the

RDS instance through the internal network. If no VPC whitelist group exists, create a new group.



8. · If you select Retain Classic Network, add the endpoint of the VPC to the ECS instance before the endpoint of the classic network expires.
   · If you clear Retain Classic Network, the internal connection from the ECS instance in the VPC to the RDS instance is immediately disconnected after the network type is changed. You must add the RDS endpoint of the VPC to the ECS instance.

> **Note:**
> To connect an ECS instance in the classic network to an RDS instance in a VPC through the internal network, you can use *ClassicLink* or switch the network type to VPC.

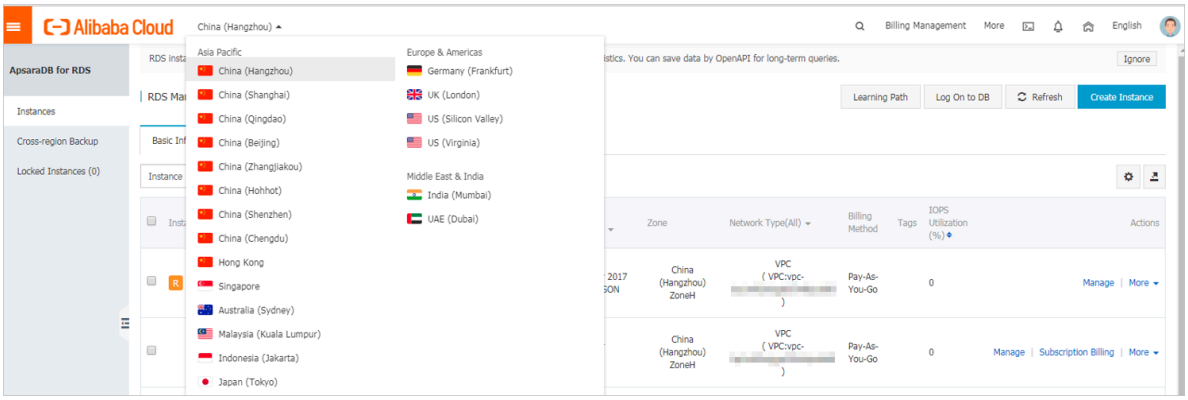
Change the expiration time for the original internal endpoint of the classic network

During the hybrid access period, you can change the retention period for the original internal endpoint of the classic network at any time as needed. The system updates the expiration date based on the modified date. For example, if the original internal endpoint of the classic network is set to expire on August 18, 2017, and you change the expiration time to "14 days later" on August 15, 2017. The internal endpoint of the classic network is released on August 29, 2017.

Follow these steps to change the expiration time:

1. Log on to the *ApsaraDB for RDS console*.

2. In the upper-left corner of the page, select the region where the instance is located.



3. Find the instance and click the instance ID.
4. In the left-side navigation pane, click Database Connections.
5. On the Instance Connection tab, click Change Expiration Time, as shown in the following figure.

6. On the Change Expiration Time page that appears, select an expiration time and click OK.

11 Database connection

11.1 Configure endpoints for an RDS for MySQL instance

This topic describes how to configure endpoints for an RDS for MySQL instance. ApsaraDB for RDS provides two types of endpoints: internal endpoints and public endpoints.

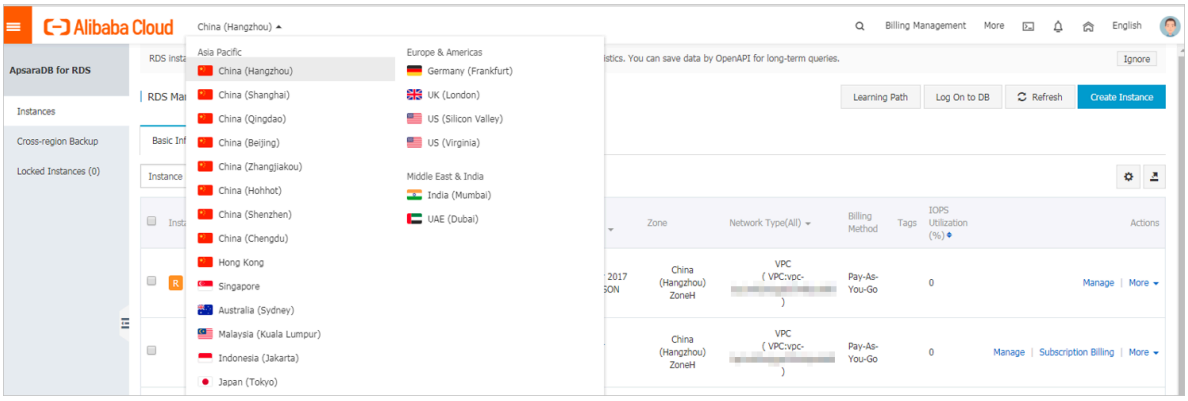
Internal and public endpoints

Endpoint type	Description
Internal endpoint	· The internal endpoint is provided by default. You do not need to apply for it and cannot release it. However, you can change the network type. · If your application is deployed to an ECS instance that is in the same region and has the same *network type* as the RDS instance, the RDS instance and the ECS instance can communicate each other by default. You do not need to apply for a public endpoint for the RDS instance. · Accessing an RDS instance through the internal endpoint achieves the high security and performance of the RDS instance.

Endpoint type	Description
Public endpoint	· You must manually apply for a public endpoint. You can release the public endpoint if you do not need it. · When you cannot access an RDS instance through the internal endpoint, you must apply for a public endpoint. The following section describes the specific scenarios: - When you access an RDS instance from an ECS instance, where the ECS instance and RDS instance are located in different regions, and the *network types* are different. - When you access an RDS instance from the third-party services or applications. **Note:** · For the security of your instance, use caution when you access RDS instances through a public endpoint. · We recommend that you migrate your application to an ECS instance in the same region and with the same network type as your RDS instance, and then use the internal endpoint to access your applications. This helps to improve transmission speed and data security.

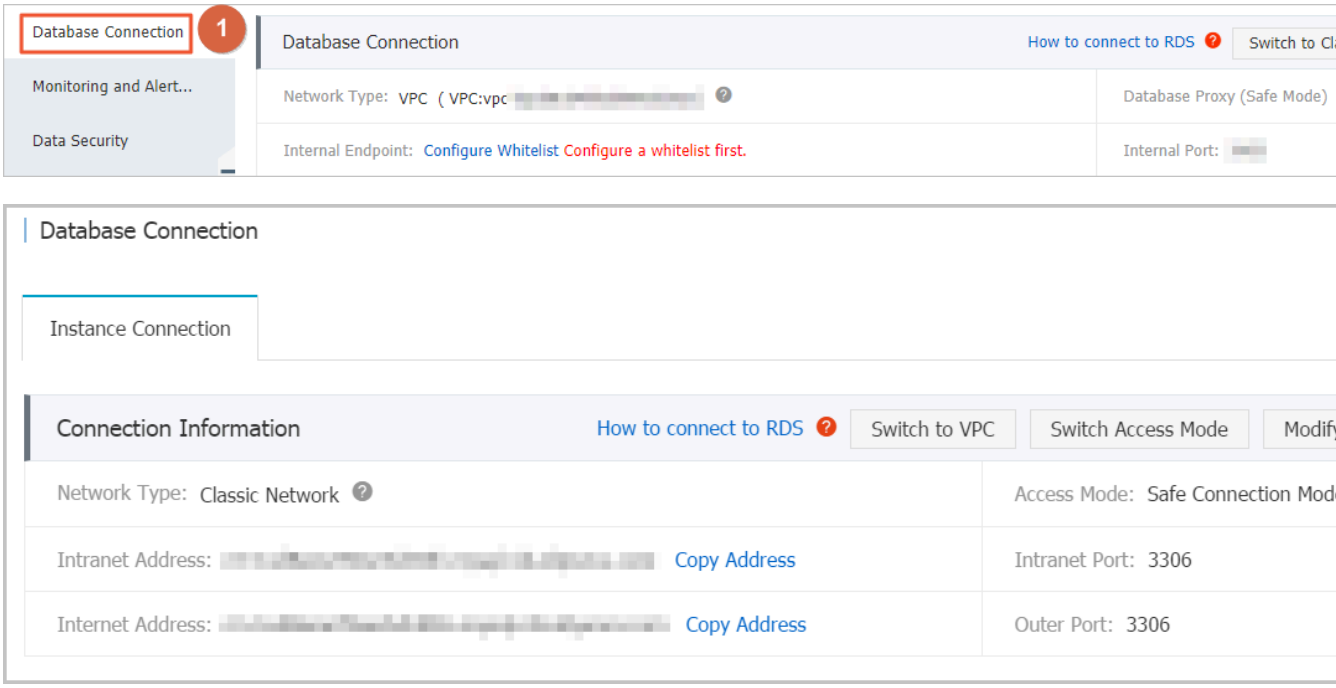
Apply for or release a public endpoint

1. Log on to the *RDS console*.
2. In the upper-left corner of the page, select the region where the instance is located.



3. Find the instance and click the instance ID.
4. In the left-side navigation pane, select Database Connections.

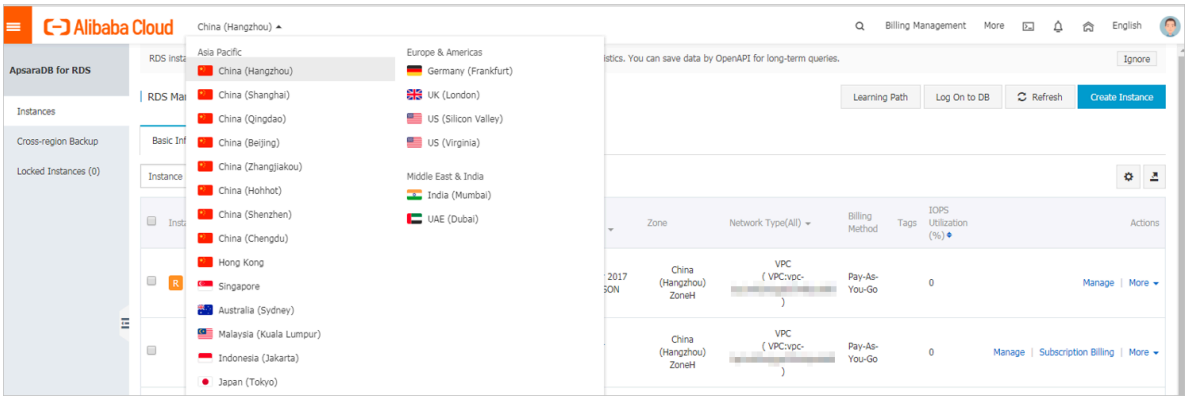
5. · If you have not applied for a public endpoint, click Apply for Public Endpoint.
   · If you have applied for a public endpoint, click Release Public Endpoint.



6. In the message box that appears, click OK.

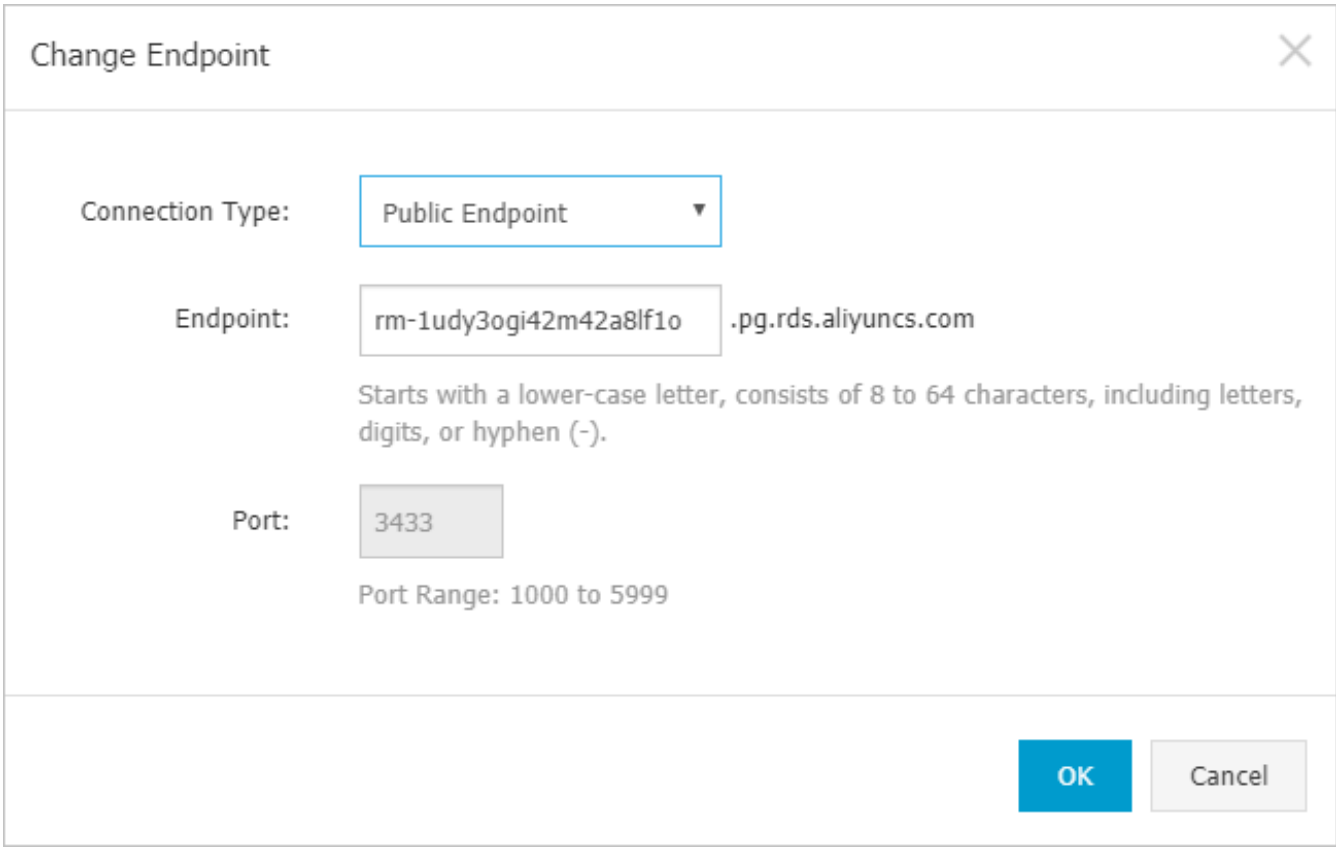
Change a public endpoint

1. Log on to the *RDS console*.
2. In the upper-left corner of the page, select the region where the instance is located.



3. Find the instance and click the instance ID.
4. In the left-side navigation pane, select `Database Connections`.
5. Click Change Endpoint.

6. In the dialog box that appears, specify the internal and public endpoints, and click OK.



Note:

- The prefix of the endpoint must be 8 to 64 characters in length and can contain letters, numbers, and hyphens (-). It must start with a lowercase letter.
- In a VPC, either the internal port number or public port number cannot be changed.
- In a classic network, either the internal port number or public port number can be changed.

APIs

Operation	Description
#unique_150	Used to apply for a public endpoint for an RDS instance.
#unique_151	Used to release the public endpoint of an RDS instance.

11.2 Log on to an RDS for MySQL instance through DMS

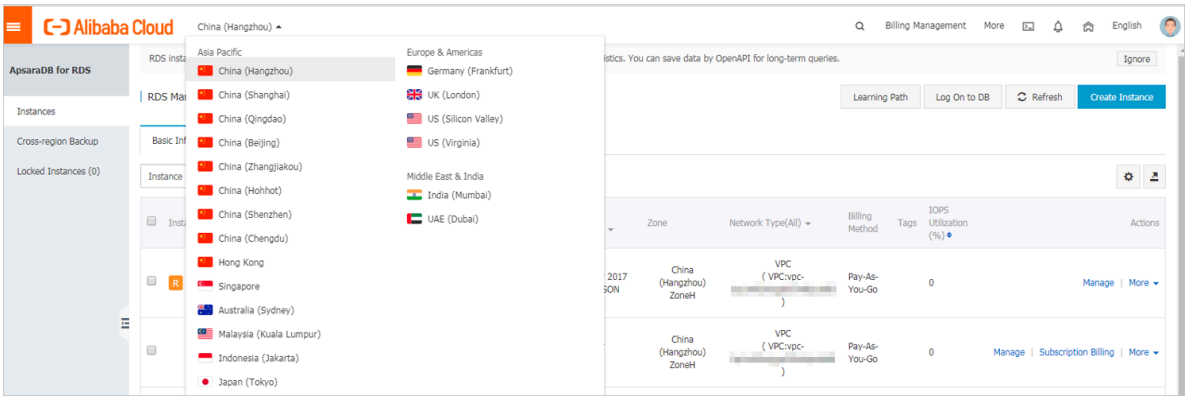
This topic describes how to log on to an RDS for MySQL instance through Alibaba Cloud Data Management Service (DMS).

Precautions

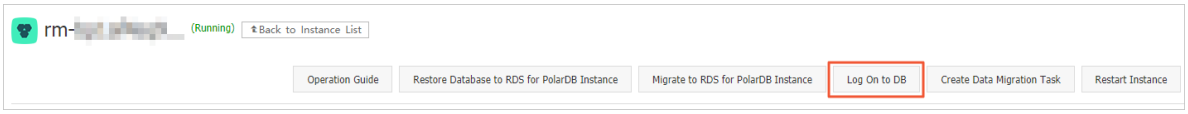
To log on to an RDS for MySQL instance through DMS, you must use the internal endpoint of the instance to establish a connection.

Procedure

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**



3. **Find the target RDS instance and click the instance ID.**
4. **In the upper-right corner, click Log On to DB.**

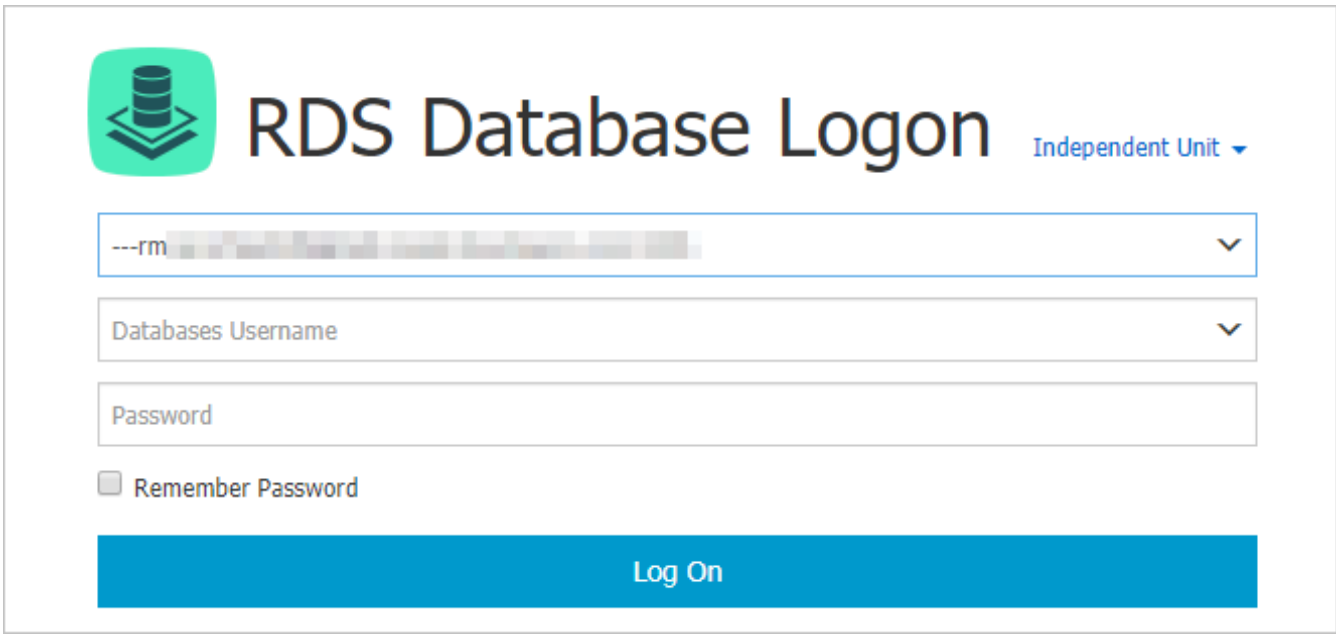


5. **Set the following parameters:**

   - **Endpoint and port: The endpoint and port for connecting the instance. The endpoint and port information is in `<Internal endpoint>:<Internal port number>` format, for example, rm-bpxxxxxxx.rds.aliyuncs.com:3433. For**

more information about how to view the endpoint and port information of the instance, see *View the internal and public endpoints and ports of an RDS for MySQL instance*.

- Databases Username: The username of the account that manages the instance.
- Password: The password of the account that manages the instance.



6. Click Log On.

> **Note:**
> If you want the browser to remember the password, select Remember Password before you click Log On.

7. If a message is displayed, asking you to add the IP address segment of the DMS server to a whitelist of the instance, click the button for setting all instances or setting the current instance.

8. After you add the IP address segment of the DMS server to a whitelist of the instance, click Log On.

11.3 Apply for a public endpoint for an RDS for MySQL instance

This topic describes how to apply for a public endpoint for an RDS for MySQL instance. Apsara for RDS supports two types of endpoints: internal endpoints and public endpoints. By default, the system provides you with an internal endpoint for connecting to your RDS instance. If you want to connect to your RDS instance through the Internet, you must apply for a public endpoint.

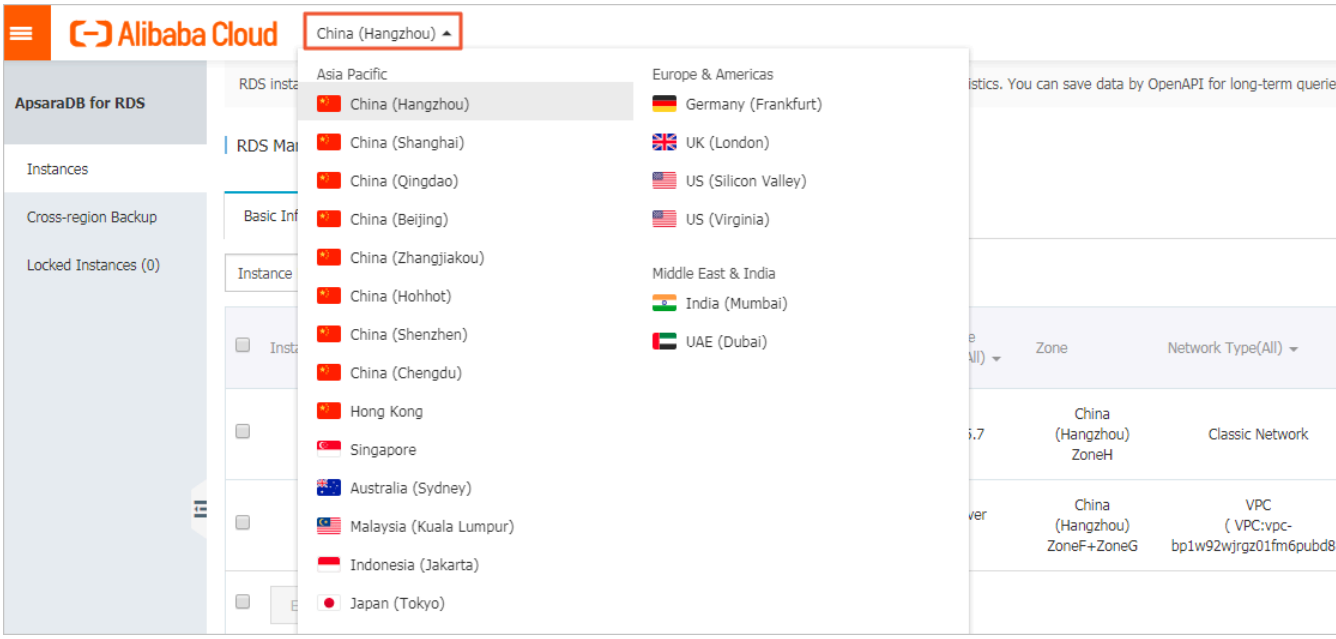
Internal and public endpoints

Endpoint type	Description
Internal endpoint	The internal endpoint is generated by default. Use the internal endpoint if all of the following conditions are met: · Your application is deployed on an ECS instance. · The ECS instance is located in the same region as your RDS instance . · The ECS instance has the same *network type* as your RDS instance. The internal endpoint is recommended because accessing RDS through the intranet is most secure and delivers optimal performance.
Public endpoint	You must manually apply for a public endpoint. You can also release it at anytime. Use the public endpoint if you cannot access RDS through the intranet . Specific scenarios are as follows: · An ECS instance accesses your RDS instance but the ECS instance is located in a different region or has a network type different from your RDS instance. · A server or computer outside Alibaba Cloud accesses your RDS instance. **Note:** · The public endpoint and traffic are currently free of charge. · Using the public endpoint reduces security. Please exercise caution. · To guarantee high security and performance, we recommend that you migrate your application to an ECS instance that is in the same region and has the same network type as your RDS instance and then use the public endpoint.

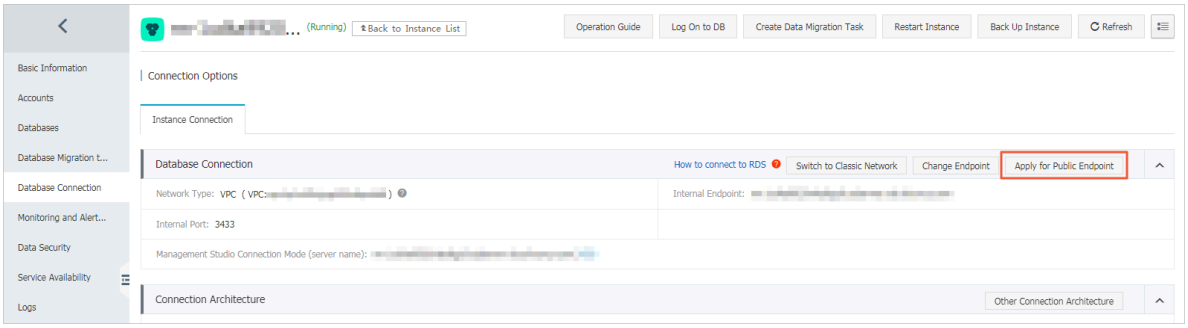
Apply for a public endpoint

1. Log on to the *RDS console*.

2. Select the target region.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Database Connection.
5. Click Apply for Public Endpoint.



6. In the displayed dialog box, click OK.

   The public endpoint is generated.

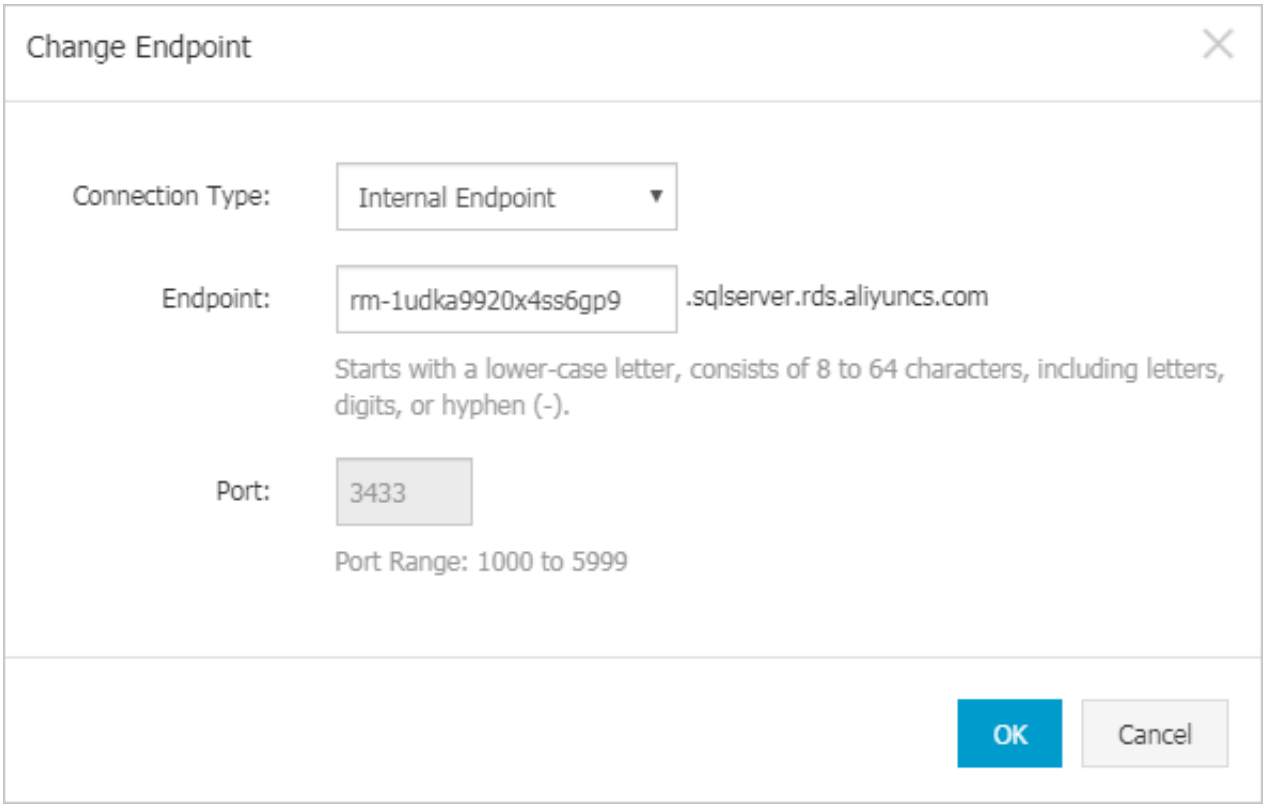
7. Optional. To modify the public endpoint or port number, click Change Endpoint. In the displayed dialog box, select a connection type, set the public endpoint and port number, and click OK.

   - Connection Type: Select Public Endpoint.

> **Note:**

The Public Endpoint option is available only after you have applied for a public endpoint.

- **Endpoint: The endpoint must be 8 to 64 characters in length and can contain letters, numbers, and hyphens (-). It must start with a lowercase letter.**
- **Port: You can change the port number only when the network type of the RDS instance is classic network.**



APIs

API	Description
#unique_150	**Used to apply for a public endpoint for an RDS instance.**

11.4 View the internal and public endpoints and ports of an RDS for MySQL instance

This topic describes how to view the internal and public endpoints and ports of an RDS for MySQL instance. When connecting to an RDS instance, you must enter its internal or public endpoint and port number.

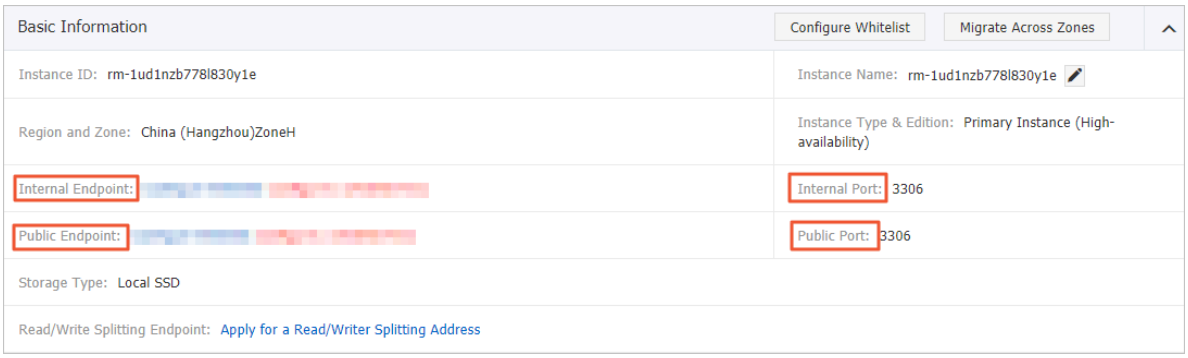
Procedure

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**

3. **Find the target RDS instance and click the instance ID.**
4. **On the Basic Information page, find the Basic Information section, where you can view the internal and public endpoints and ports of the RDS instance.**

Note:

- **The internal and public endpoints are displayed only after you configure a whitelist.**
- **The public endpoint is displayed only after you apply for it.**



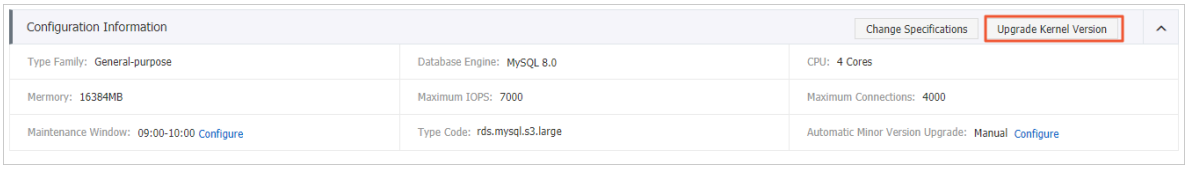
12 Dedicated database proxy

This topic describes the dedicated database proxy service of ApsaraDB for RDS. This service provides functions including read/write splitting and shot-lived connection optimization.

Prerequisites

- **The DB engine version and edition are MySQL 8.0 High-availability Edition (with local SSDs).**
- **The kernel minor version is 20190915 or later.**

> **Note:**
> **You can navigate to the Basic Information page on the RDS console and then in the Configuration Information section check for the Upgrade Kernel Version button. If this button is displayed, you can click it to view the kernel minor version. If this button is not displayed, the latest kernel minor version is used. For more information, see** *Upgrade the kernel minor version of an RDS for MySQL instance*.



- The RDS instance belongs to the following regions and zones:
  - China (Hangzhou): Zones E, F, H, and I
  - China (Shanghai): Zones A, B, D, and F
  - China (Beijing): Zones A, B, C, D, and H
  - China (Zhangjiakou): Zones A and B
  - China (Hohhot): Zones A and B
  - China (Shenzhen): Zones A, B, C, D, and E
  - Singapore: Zones A and C
  - India (Mumbai): Zones A and B
  - Germany (Frankfurt): Zone A
  - Japan (Tokyo): Zones A and B
  - Australia (Sydney): Zones A and B
  - UK (London): Zones A and B

  Note:
  If you want to enable this service for an RDS instance that does not belong to the preceding regions and zones, you must *open a ticket*.

Context

This service uses dedicated proxy compute resources to establish a dedicated database proxy for your RDS instance. It has the following benefits:

- A proxy endpoint is used and remains valid unless you release the proxy instance. Therefore, you do not need to frequently change the endpoint information in your application. For example, you do not need to do so if you enable read/write splitting but later you release the read-only instances attached to the RDS instance and disable read/write splitting. This reduces maintenance costs.
- Dedicated database proxy guarantees service stability. If users share the same database proxy, they may compete for resources, resulting in service instability.
- Resources can be scaled to process more traffic.

Billing

In the beta test phase, this service is free of charge. After the beta test phase ends, you will be notified of when you must pay for this service.

Precautions

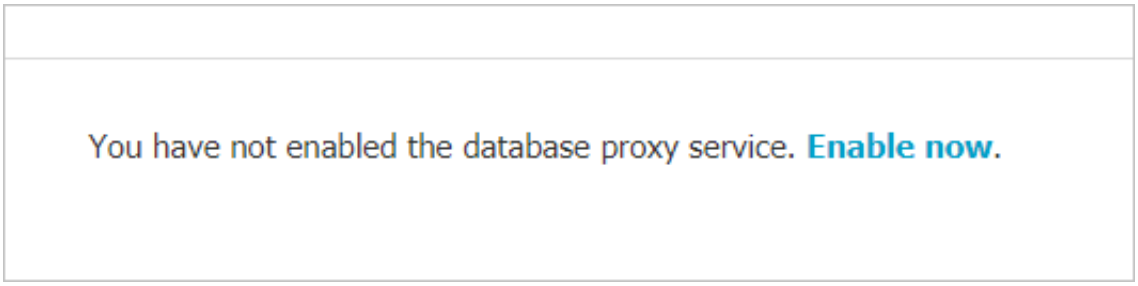
- **A transient disconnection may occur when you change the configuration of the master instance or its read-only instance.**
- **The database proxy service does not support SSL encryption.**
- **The database proxy service does not support compression protocols.**
- **When the proxy endpoint is used, all transaction requests are routed to the master instance.**
- **When the proxy endpoint is used for read/write splitting, the read consistency of non-transaction requests cannot be guaranteed. If you require read consistency, then you can encapsulate data in transactions.**
- **When the proxy endpoint is used, the `/*FORCE_MASTER*/` class cannot be used to comment SQL statements to query the master instance.**
- **When the proxy endpoint is used, the `LAST_INSERT_ID()` function cannot be used.**
- **When the proxy endpoint is used, the `show processlist` command returns a combination of results from the master instance and all its read-only instances.**
- **If short-lived connection optimization is enabled, the `show processlist` command may return idle connections.**
- **If you run *multiple statements* or stored procedures, all subsequent requests over the current connection are routed to the master instance. To restore read/write splitting, you must disconnect the current connection and then reestablish a connection.**

Enable database proxy

1. **Log on to the *RDS console*.**

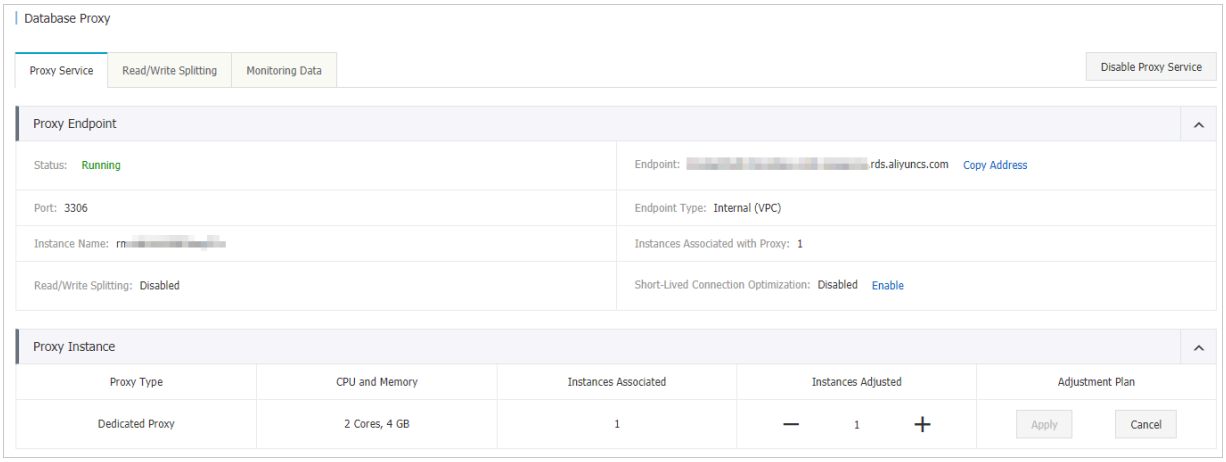
2. **In the upper-left corner, select the region where the target RDS instance is located.**

3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Database Proxy.**
5. **In the main workspace, click Enable now.**



Overview of the Database Proxy page

After database proxy is enabled, you can use the proxy endpoint to enable read/write splitting and short-lived connection optimization.



Category	Parameter	Description
Proxy Endpoint	Endpoint	The proxy endpoint provided by the dedicated database proxy. The read/write splitting function is bound to the proxy endpoint.
	Port	The port used by the proxy endpoint.
	Endpoint Type	Only the Internal (VPC) network type is supported.
	Instance Name	The RDS instance connected through the proxy endpoint.
	Instances Associated with Proxy	The number of proxy instances associated with the proxy endpoint. You can add proxy instances to increase the request-processing capability of the dedicated database proxy. After the beta test phase ends, you must pay for the proxy instances you add.
	Read/Write Splitting	Whether read/write splitting is enabled for the proxy endpoint.
	Short-Lived Connection Optimization	Whether short-lived connection optimization is enabled for the proxy endpoint. **Note:** You can click Enable or Disable to the right of Short-Lived Connection Optimization to enable or disable this function.
Proxy Instance	Proxy Type	Only the Dedicated Proxy service is supported.
	CPU and Memory	The specification of the proxy instance. Only the 2 Cores, 4 GB specification is supported.
	Instances Associated	The number of proxy instances associated with the proxy endpoint.

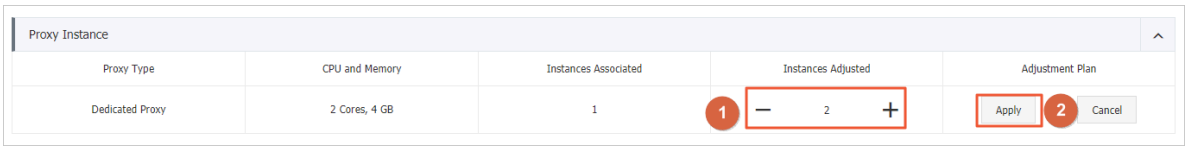
Change the number of proxy instances

Note:
Changing the number of proxy instances causes a transient disconnection. You must make sure that your application can automatically reconnect to the RDS instance.

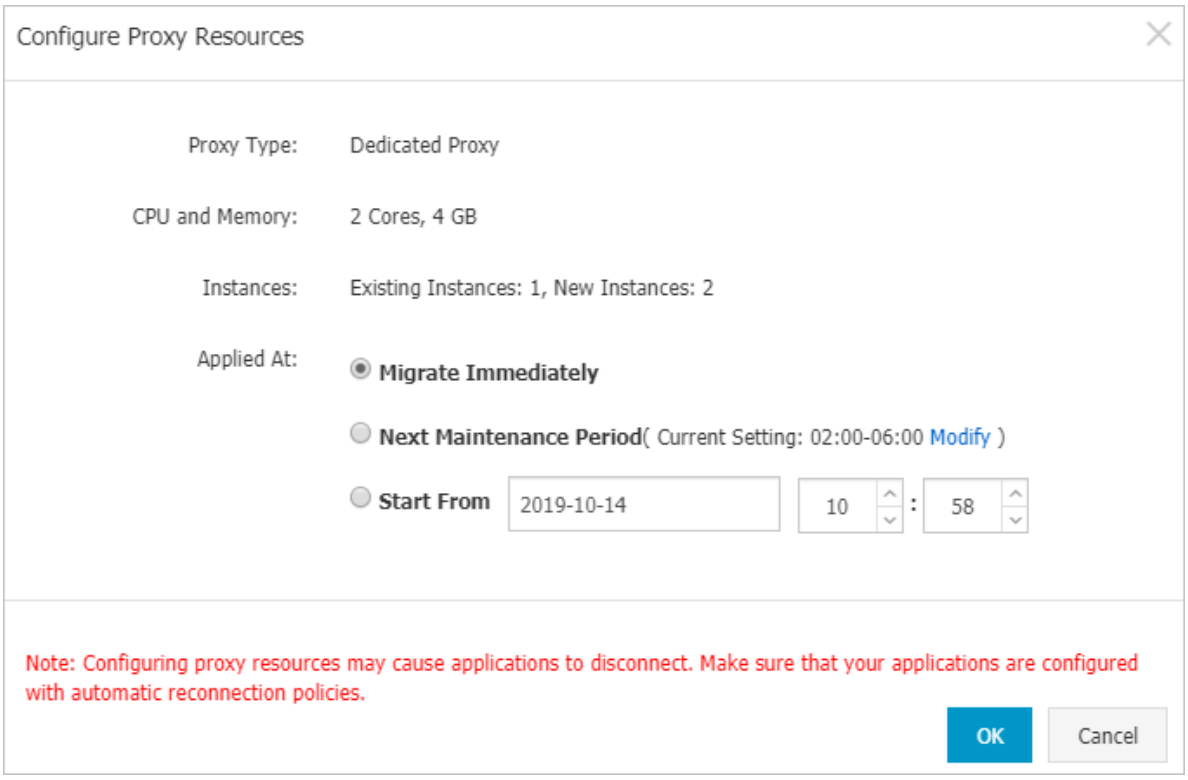
1. **Log on to the *RDS console*.**

2. In the upper-left corner, select the region where the target RDS instance is located.

3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Database Proxy.
5. On the Proxy Service tab, find the Proxy Instance section. Then change the number in the Instances Adjusted column and click Apply in the Adjustment Plan column.

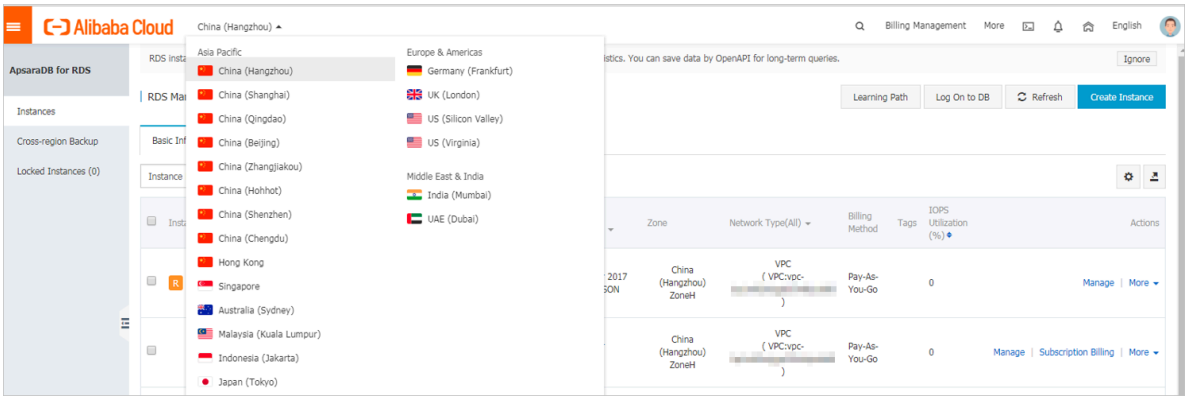


6. Set Applied At and click OK.



View monitoring data

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**



3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Database Proxy.**
5. **Select the Monitoring Data tab.**
6. **Specify a time range and view the CPU Utilization (%) metric.**

> **Note:**
> **The CPU Utilization (%) metric indicates the CPU usage of the physical host where the proxy instance is located.**

Disable database proxy

1. **Log on to the *RDS console*.**

2. In the upper-left corner, select the region where the target RDS instance is located.

3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Database Proxy.
5. In the upper-right corner, click Disable Proxy Service.

6. In the displayed dialog box, click OK.

APIs

API	Description
#unique_156	Used to modify the settings of the dedicated database proxy service for an RDS instance.
#unique_157	Used to enable or disable the dedicated database proxy service for an RDS instance.
#unique_158	Used to query details about the dedicated database proxy service for an RDS instance.
#unique_159	Used to query the proxy endpoint of an RDS instance.
#unique_160	Used to modify the proxy endpoint of an RDS instance.

API	Description
#unique_161	**Used to query the performance data of the dedicated database proxy for an RDS instance.**

13 Monitoring and alerts

13.1 View resource and engine monitoring data

This topic describes how to view the resource and engine monitoring data of an RDS for MySQL instance. ApsaraDB for RDS provides a wide range of performance metrics for you to view in the RDS console.

CloudDBA provides intelligent diagnosis and optimization for monitoring services.

Procedure

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**

3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Monitoring and Alerts.**

5. On the Monitoring tab, select the Resource Monitoring or Engine Monitoring monitoring type and specify the time range. The following table describes the monitoring metrics.

Monitoring type	Metric	Description
Resource Monitoring	Disk Space (MB)	The disk space usage of the RDS instance, including: · Instance Size · Data Usage · Log Size · Temporary File Size · Other System File Size Unit: MByte.
	IOPS (Input/ Output Operations per Second)	The number of I/O requests per second for the RDS instance. Unit: Number/second.
	Total Connections	The total number of connections to the RDS instance, including the number of active connections and the total number of connections.
	CPU and Memory Usage (%)	The CPU and memory usage of the RDS instance (excluding the CPU and memory usage for the operating system).
	Network Traffic (KB)	The input and output traffic of the RDS instance per second. Unit: KB.
Engine Monitoring	TPS (Transactions per Second)/ QPS (Queries per Second)	The average number of transactions per second and the average number of SQL statements executed per second.
	InnoDB Buffer Pool Read Hit Ratio, Usage Ratio, and Dirty Block Ratio (%)	The read hit ratio, usage, and proportion of dirty blocks for the InnoDB buffer pool.
	InnoDB Read/ Write Volume (KB)	The amount of data that is read and written by InnoDB per second. Unit: KB.

Monitorin type	Metric	Description
	InnoDB Buffer Pool Read/Write Frequency	The number of read and write operations that are performed by InnoDB per second.
	InnoDB Log Read/ Write/fsync	The average frequency of physical writes to log files per second by InnoDB, the log write request frequency, and the average frequency of fsync () writes to log files.
	Number of Temporary Tables Created Automatically on the Hard Disk when MySQL Statements Are Being Executed	The number of temporary tables that are automatically created on the hard disk when the instance runs SQL statements.
	MySQL_COMDML	The number of SQL statements that are executed by the RDS instance per second, including: · Insert · Delete · Insert_Select · Replace · Replace_Select · Select · Update
	MySQL_RowDML	The number of operations that are performed by InnoDB per second, including: · The average number of physical writes to log files per second · The number of rows that are read/updated /deleted/inserted from InnoDB tables per second.
	MyISAM Read/ Write Frequency	The number of read/write operations that are performed by MyISAM on the buffer pool per second and the number of read/write operations that are performed by MyISAM on the hard disk per second.

Monitorin type	Metric	Description
	MyISAM Key Buffer Read/Write /Usage Ratio (%)	The read hit ratio, write hit ratio, and usage of the MyISAM key buffer per second.

13.2 Set the monitoring frequency

This topic describes how to set the monitoring frequency for an RDS instance. ApsaraDB for RDS provides more than one monitoring frequency for you to select.

Background information

Two monitoring frequencies provided before May 15, 2018

- **Once per 60 seconds (monitoring period: 30 days)**
- **Once per 300 seconds (monitoring period: 30 days)**

Second-level monitoring frequency introduced since May 15, 2018

Minute-level monitoring frequencies cannot meet monitoring requirements of some users and maintenance personnel. Therefore, since May 15, 2018, RDS has introduced second-level monitoring frequencies. This facilitates problem locating and improves customer satisfaction.

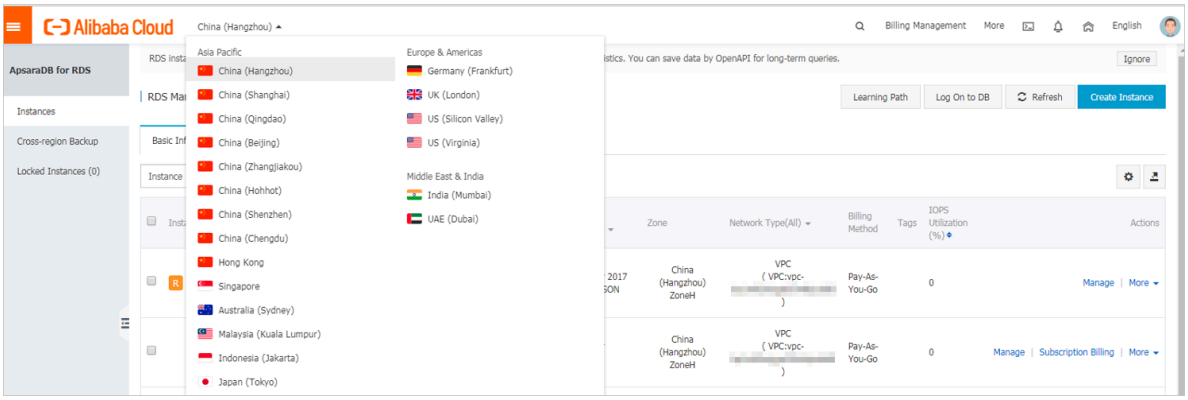
Once per 5 seconds (monitoring period: 7 days), turning to once per minute since the eighth day

The detailed monitoring policies are described in the following table.

Instance type	Once per 5 seconds	Once per minute (60 seconds)	Once per 5 minutes (300 seconds)
Basic Edition	Not supported	Supported for free	Default configuration
High-availability or Enterprise Edition : Memory < 8 GB	Not supported	Supported for free	Default configuration
High-availability or Enterprise Edition : Memory ⩾ 8 GB	Supported (Not free)	Default configuration	Supported for free

Procedure

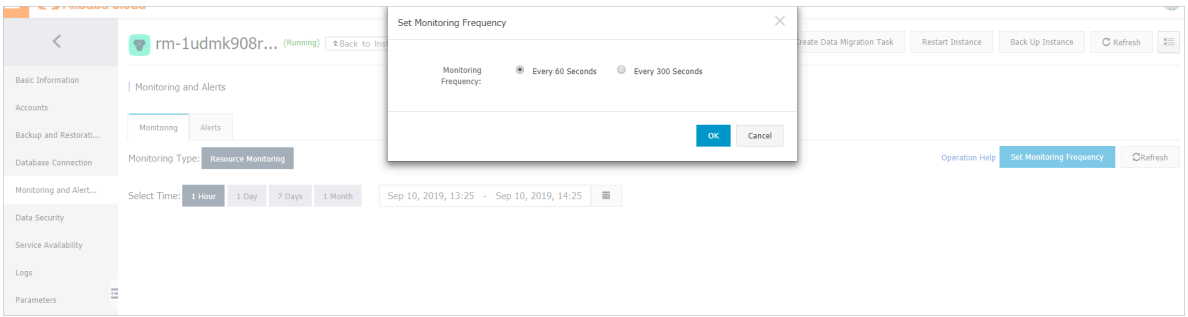
1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**



3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Monitoring and Alerts.**

> **Note:**
> **For information about the monitoring metrics supported by the instance, see *View resource and engine monitoring data*.**

5. **Click the Monitoring tab.**
6. **Click Set Monitoring Frequency.**
7. **In the Set Monitoring Frequency dialog box, select the monitoring frequency and click OK.**



> **Note:**
> **If the instance does not support second-level monitoring, a message is displayed to tell you so in the Set Monitoring Frequency dialog box.**

APIs

API	Description
#unique_165	Used to query the monitoring data of an RDS instance.

13.3 Set an alert rule

This topic describes how to set an alert rule for an RDS instance. ApsaraDB for RDS offers the instance monitoring function, and sends messages to you after detecting an exception in an instance. In addition, when the instance is locked due to insufficient disk space, the system sends a message to you.

Background information

Alibaba CloudMonitor offers monitoring and alarming. CloudMonitor helps you set alarm rules for metrics. You must add alarm contacts while set a contact group. The alarm contacts and the contact group are notified immediately when an alarm is triggered in the event of exceptions. You can create an alarm contact group using a related metric.

Procedure

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**

3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Monitoring and Alerts.**
5. **Click the Alerts tab.**

6. Click Set Alert Rule.

   You are directed to the CloudMonitor console.

   > **Note:**
   > You can click Refresh to manually refresh the current status of the alert metrics.

7. In the left-side navigation pane, choose Alarms > > Alarm Contacts to open the Alarm Contact Management page.

   > **Note:**
   > When alert rules are set for the first time, if the alert notification object is not a contact of the Alibaba Cloud account of RDS, the alarm contact and alarm contact group must be created first. If you have already set the alarm contact and the alarm contact group, go to Step 10.

8. Click Create Alarm Contact.
9. In the Set Alarm Contact dialog box, enter the alarm contact information and click Send verification code. Then, enter the verification code sent to your mailbox, and click Save.

   > **Note:**
   > - We recommend that you perform the next step to create the alarm contact group after you add all alarm notification objects.
   > - You can click Edit to modify a contact, or click Delete to delete a contact.

10. On the Alarm Contact Management page, click the Alarm Contact Group tab.
11. Click Create Alarm Contact Group.
12. Set `Group Name` and `Description`, select a contact from Existing Contacts, click to add the contact to `Selected Contacts`, and click OK.

    > **Note:**
    > On the Alarm Contact Group page, you can click to modify a contact group, click X to delete a contact group, or click Delete to delete a contact in the contact group.

13. After creating the alarm contact group, choose Cloud Service Monitoring > ApsaraDB for RDS from the left-side navigation pane.
14. Select the region of RDS for which the alert rule is to be set.
15. Find the target instance and click Alarm Rules in the Actions column.

    The system displays the metrics of the current alert.
16. Click Create Alarm Rule to add a new alert rule.

Note:

You can click Modify, Disable, or Delete for the metrics as needed.

14 Data security

14.1 Switch to the enhanced whitelist mode for an RDS for MySQL instance

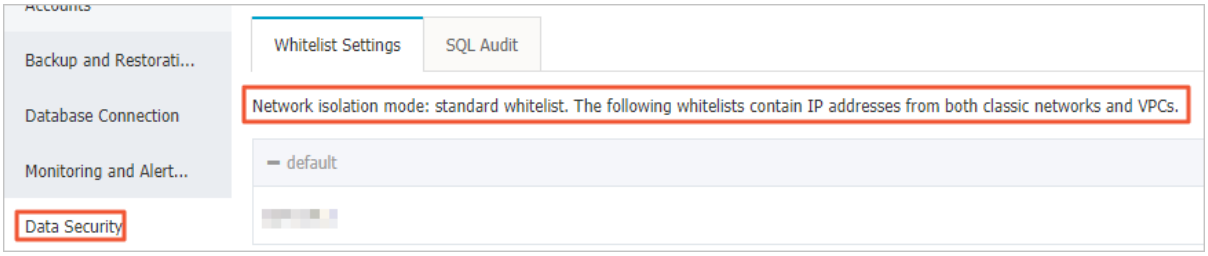
This topic describes how to switch from the standard whitelist mode to the enhanced whitelist mode for an RDS for MySQL instance.

IP whitelist modes

ApsaraDB for RDS instances provide the following two IP whitelist modes:

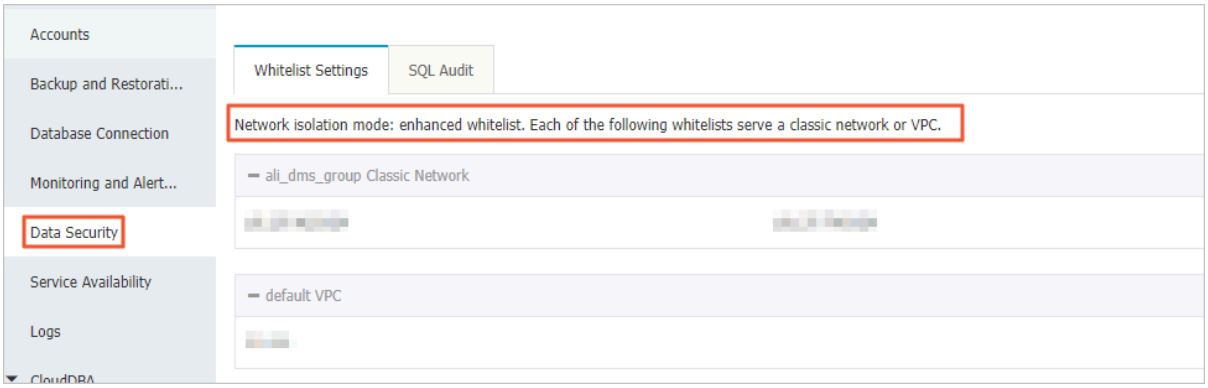
- **Standard whitelist mode**

  In this mode, the IP addresses in the whitelist do not distinguish between classic networks and VPCs. The IP addresses in the whitelist can access the RDS instance both in classics network and VPCs. We recommend that you switch from the standard whitelist to the enhanced whitelist.



- **Enhanced whitelist mode**

  In this mode, the whitelist is classified into two IP whitelist groups by network type: the classic-network whitelist group and the VPC whitelist group. When you create an IP whitelist, you must specify a network type.



Changes after switching to the enhanced whitelist mode

- **If the network type of the instance is VPC, a new whitelist of the VPC is generated and contains the same IP addresses in the original whitelist. The new IP whitelist group only applies to VPCs.**
- **If the instance network type is classic network, a new whitelist group is generated and contains the same IP addresses in the original whitelist. The new IP whitelist group only applies to classic networks.**
- **If the instance is in the *hybrid access mode*, two new whitelist groups are generated and each contains the same IP addresses in the original whitelist. One of the whitelist group applies to VPCs and the other applies to classic networks.**

> **Note:**
> **Switching to the enhanced whitelist mode does not affect the ECS instances that are in the *ECS security group whitelist*.**

Prerequisites

The DB version used by the instance is one of the following:

- **MySQL 5.7 High-Availability Edition**
- **MySQL 5.6**
- **MySQL 5.5**

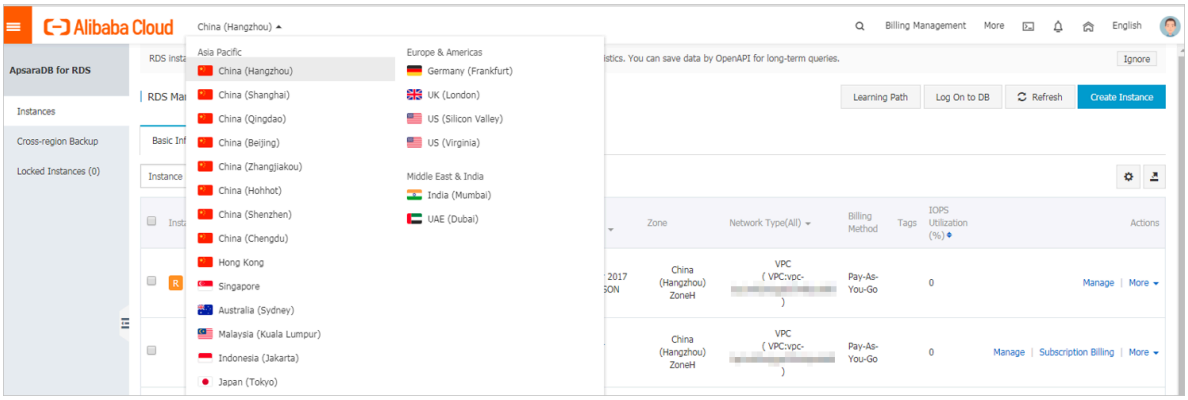
Precautions

- **You can switch from the standard whitelist mode to the enhanced whitelist mode. However, you cannot switch from the enhanced whitelist mode to the standard whitelist mode.**
- **In the enhanced whitelist mode, the classic-network whitelist group also applies to accesses from a public network. If you want to access the RDS instance from an instance, host, or application in the public network, you must add the public IP address to the classic-network whitelist group.**

Procedure

1. **Log on to the *RDS console*.**

2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Data Security.
5. On the Whitelist Settings tab, click Switch to Enhanced Whitelist (Recommended).

6. In the message box that appears, click OK.

14.2 Configure a whitelist for an RDS for MySQL instance

This topic describes how to configure a whitelist for an RDS for MySQL instance.

RDS for PostgreSQL provides two types of whitelists::

- IP address whitelist: Add IP addresses to a whitelist so that these IP addresses can access the RDS instance.

> **Note:**
> The default whitelist is an IP address whitelist that contains only the default IP address 127.0.0.1. This default IP address means that no devices can access the RDS instance.

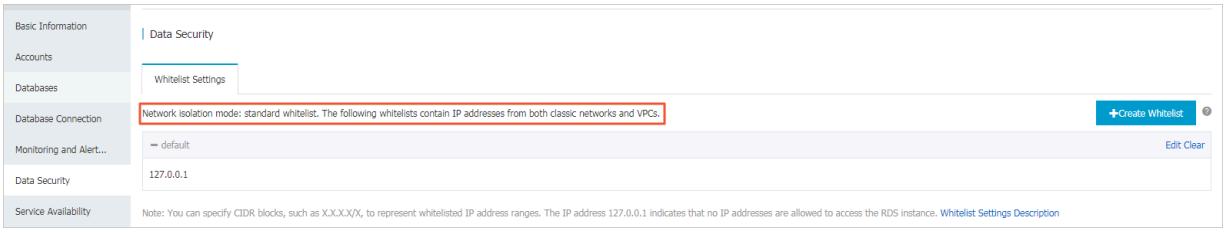
- **VPC security group whitelist: Add a VPC security group to a whitelist so that all ECS instances in the VPC security group can access the RDS instance.**

A whitelist can be used to improve the security of your RDS instance. We recommend that you periodically check and adjust your whitelists to maintain RDS security. Configuring a whitelist does not affect the normal running of the RDS instance.

Configure an IP address whitelist

Precautions

- **The default IP whitelist can only be edited or cleared, but cannot be deleted.**
- **If you log on to DMS but the DMS server IP address has not been added to the whitelist, DMS prompts you to add the IP address and automatically generates a whitelist containing the IP address.**
- **You must confirm which network isolation mode your RDS instance is in before configuring the whitelist. See the corresponding operations based on the network isolation mode.**



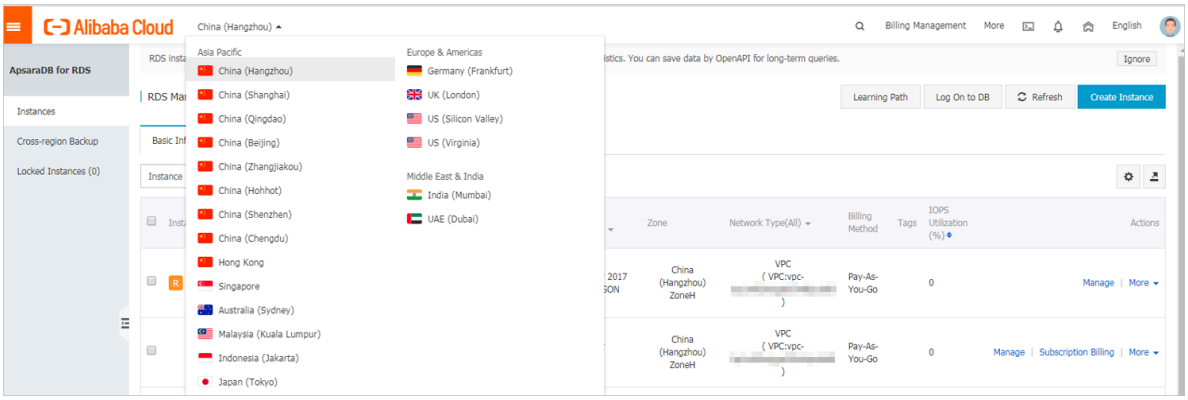
Note:

The private networks to which RDS instances belong are divided into two types: Classic Network and VPC.

- **Classic network: Alibaba Cloud automatically allocates IP addresses. Users only need to perform simple configurations. This network type is suitable for new users.**
- **VPC: You can customize the network topology and IP addresses. It supports leased line connection, and is suitable for advanced users.**

If the network isolation mode is enhanced whitelist mode

1. **Log on to the *RDS console*.**
2. **In the upper-left corner of the page, select the region where the target instance is located.**



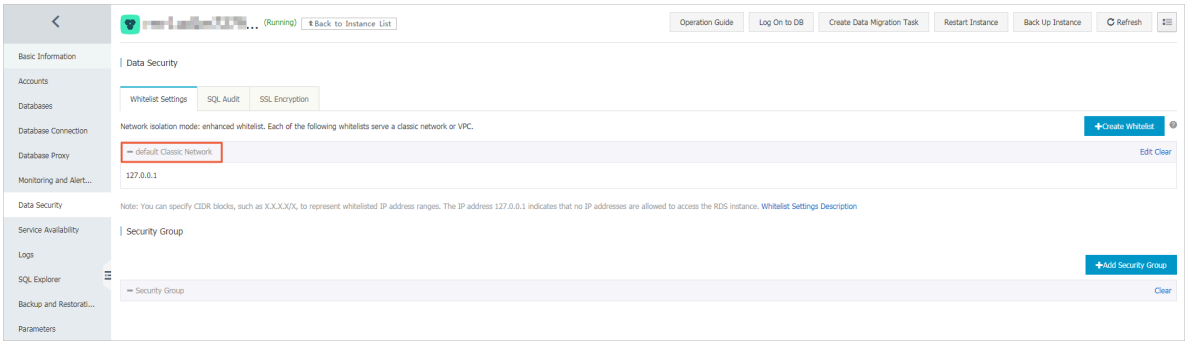
3. **Find the target instance and click its ID.**
4. **In the left-side navigation pane, click Data Security.**
5. **On the Whitelist Settings tab page, follow these instructions based on your scenario:**

   - **To access the RDS instance from an ECS instance located within a VPC, click Edit for `default VPC` whitelist.**
   - **To access the RDS instance from an ECS instance located within the classic network, click Edit for the `default Classic Network` whitelist.**
   - **To access the RDS instance from a server or computer located in a public network, click Edit for the `default Classic Network` whitelist.**

Note:

- **To allow ECS to access RDS through the intranet (VPC or classic network), make sure that the two instances are in the same region and have the same *network type*. Otherwise, the connection fails.**

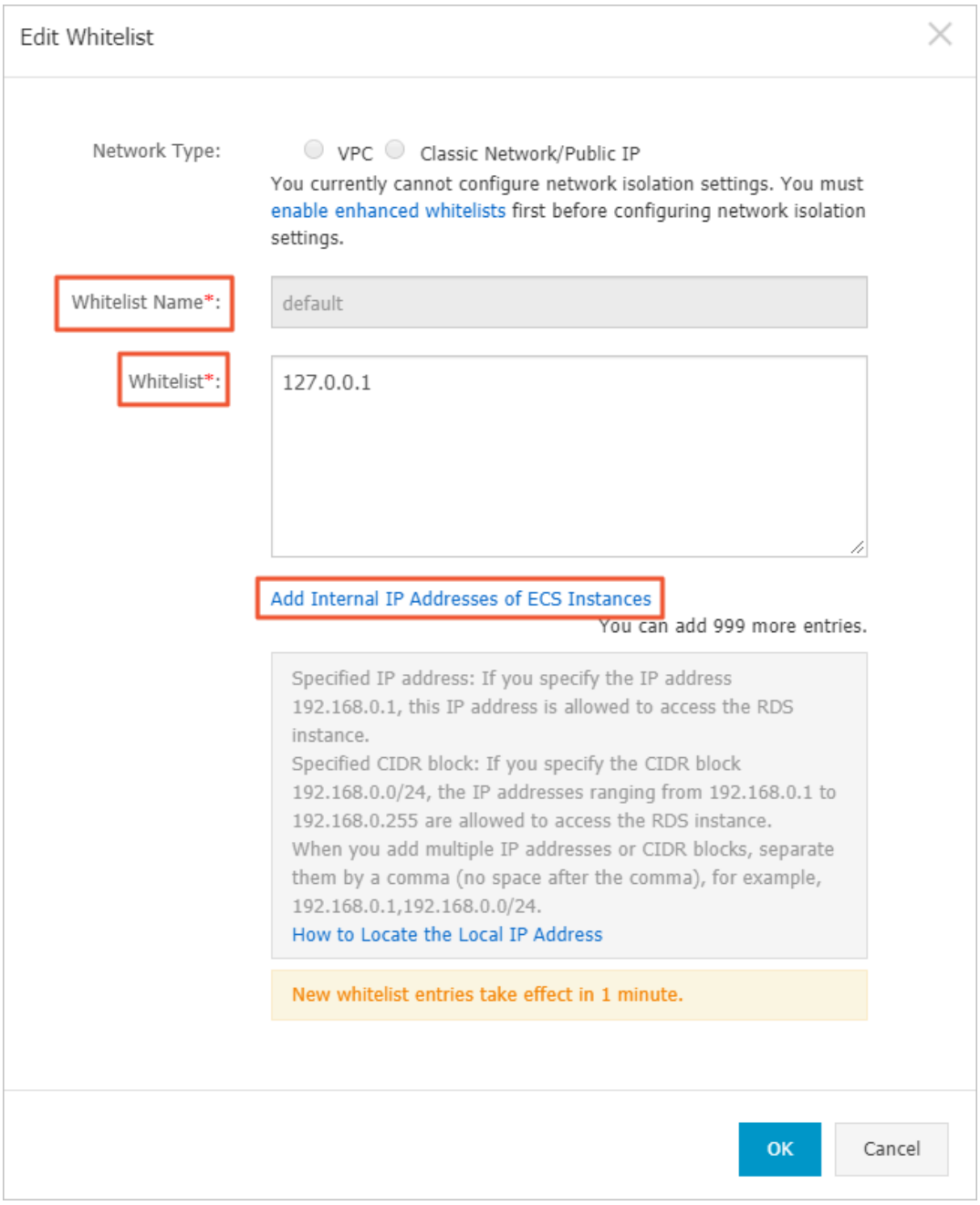
- You can also click Create Whitelist to create a new whitelist. In the displayed Create Whitelist dialog box, select a network type: VPC or Classic Network/ Public IP.



6. In the displayed Edit Whitelist dialog box, add the IP addresses that need to access the instance, and then click OK.
   - If you add an IP address range, such as 10.10.10.0/24, any IP address in the 10.10.10.*X* format can access the RDS instance.
   - To add multiple IP addresses or IP address ranges, separate them with a comma (without spaces), for example, `192.168.0.1,172.16.213.9`.
   - You can click Add Internal IP Addresses of ECS Instances to display the IP addresses of all the ECS instances under your Alibaba Cloud accoun. and add to the whitelist.

Note:

After you add any IP address to the default whitelist, the default IP address 127.0.0.1 is automatically removed.

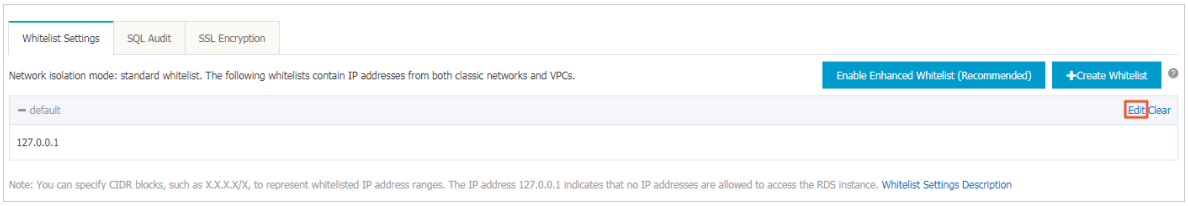


If the network isolation mode is standard whitelist mode

1. **Log on to the *RDS console*.**
2. **In the upper-left corner of the page, select the region where the target instance is located.**
3. **Find the target instance and click its ID.**
4. **In the left-side navigation pane, click Data Security.**

5. On the Whitelist Settings tab page, click Edit for the default whitelist.

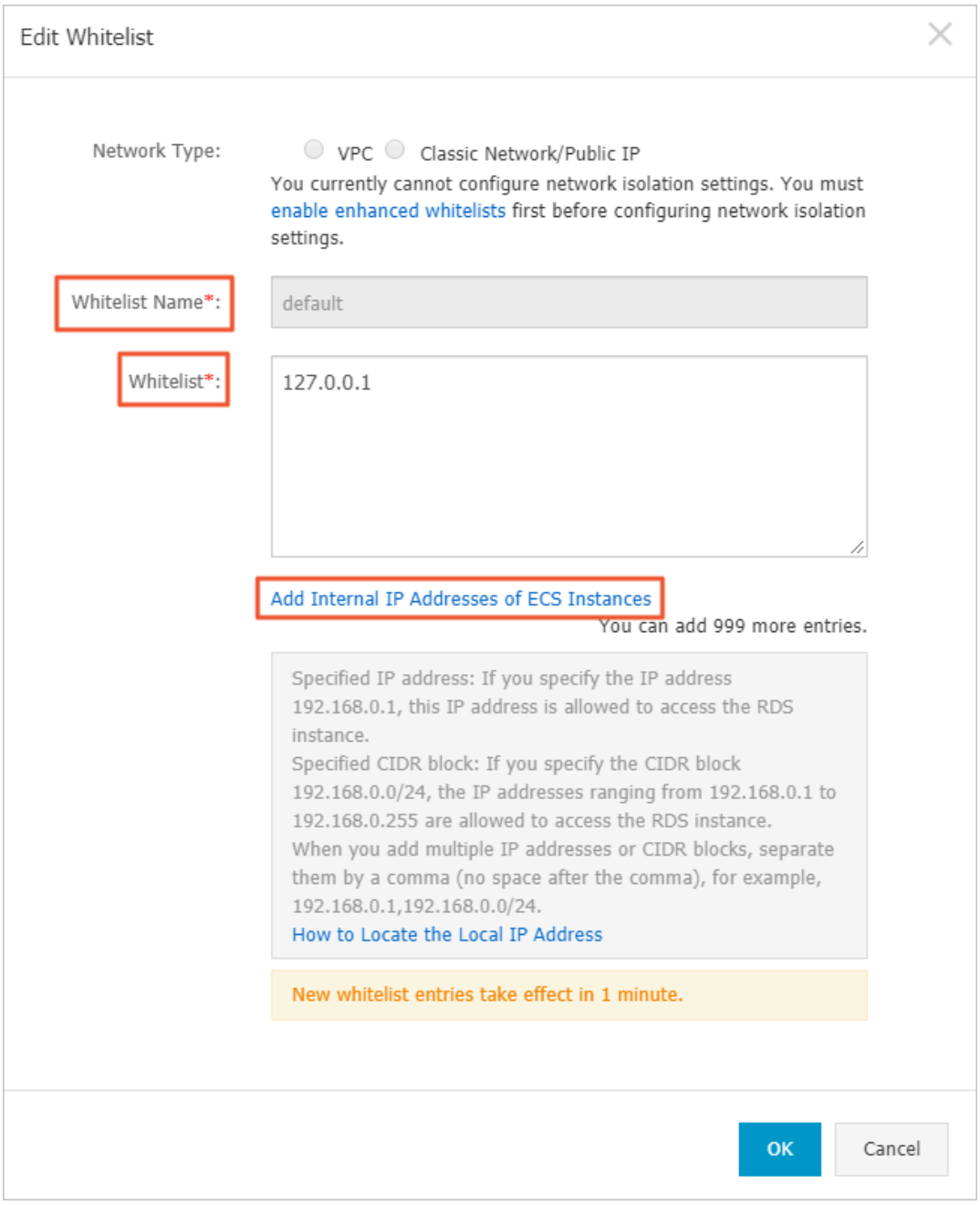
> **Note:**
> You can also click Create Whitelist to create another whitelist.



6. In the displayed Edit Whitelist dialog box, add the IP addresses that need to access the instance, and then click OK.
   - If you add an IP address range, such as 10.10.10.0/24, any IP address in the 10.10.10.*X* format can access the RDS instance.
   - To add multiple IP addresses or IP address ranges, separate them with a comma (without spaces), for example, `192.168.0.1,172.16.213.9`.
   - You can click Add Internal IP Addresses of ECS Instances to display the IP addresses of all the ECS instances under your Alibaba Cloud accoun. and add to the whitelist.

> **Note:**

After you add any IP address to the default whitelist, the default address 127.0.0.1 is automatically removed.



Common incorrect settings

- **If the IP address whitelist contains only the default address 127.0.0.1, no device is allowed to access the RDS instance. Therefore, you must add IP addresses of devices to the whitelist to allow access to the instance.**
- **The IP address in the whitelist is 0.0.0.0, but the correct format is 0.0.0.0/0.**

Note:

0.0.0.0/0 indicates that all devices are allowed to access the RDS instance. Exercise caution when using this IP address.

- **If the *enhanced whitelist* mode is enabled, check the following:**
  - **To connect to the RDS instance through the VPC address of the RDS instance, make sure that the private IP address of the ECS instance is added to the default VPC whitelist.**
  - **To connect to the RDS instance through the classic network address of the RDS instance, make sure that the private IP address of the ECS instance is added to the default Classic Network whitelist.**
  - **To connect to the RDS instance through *ClassicLink*, make sure that the private IP address of the ECS instance is added to the default VPC whitelist.**
  - **To connect to the RDS instance through the public address of the RDS instance, make sure that the public IP address of the device is added to the default Classic Network whitelist.**
- **The public IP address that you add to the whitelist may not be the real outbound IP address. The reasons are as follows:**
  - **The public IP address is not fixed and may dynamically change.**
  - **The tools or websites used to query the public IP addresses may provide wrong IP addresses.**

  For more information, see *#unique_170*.

Configure an ECS security group

An ECS security group is a virtual firewall that is used to control the inbound and outbound traffic of ECS instances in the security group. After an ECS security group is added to the RDS whitelist, the ECS instances in the security group can access the RDS instance.

For more information, see *Create a security group*.

Precautions

- **RDS versions that support an ECS security group are MySQL 5.6 and 5.7.**
- **You can configure both the IP address whitelists and an ECS security group. The IP addresses in the whitelists and the ECS instances in the security group can all access the RDS instance.**
- **You can only add one ECS security group to an RDS instance.**

- **Updates to the ECS security group are automatically synchronized to the whitelist in real time.**

Procedure

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target instance is located.**
3. **Find the target instance and click its ID.**
4. **In the left-side navigation pane, click Data Security.**
5. **On the Whitelist Settings tab page, click Add Security Group.**

   Note:
   Security groups with a VPC tag are within VPCs.
6. **Select a security group and click OK.**

APIs

API	Description
#unique_171	**Used to view the IP address whitelist of an RDS instance.**
#unique_172	**Used to modify the IP address whitelist of an RDS instance.**

14.3 Configure SSL encryption for an RDS for MySQL instance

This topic describes how to enable Secure Sockets Layer (SSL) encryption and install SSL CA certificates to applications. SSL encrypts data over network connections at the transport layer. This enhances data security and integrity but increases network connection response time.

Prerequisites

The DB engine version and edition are one of the following:

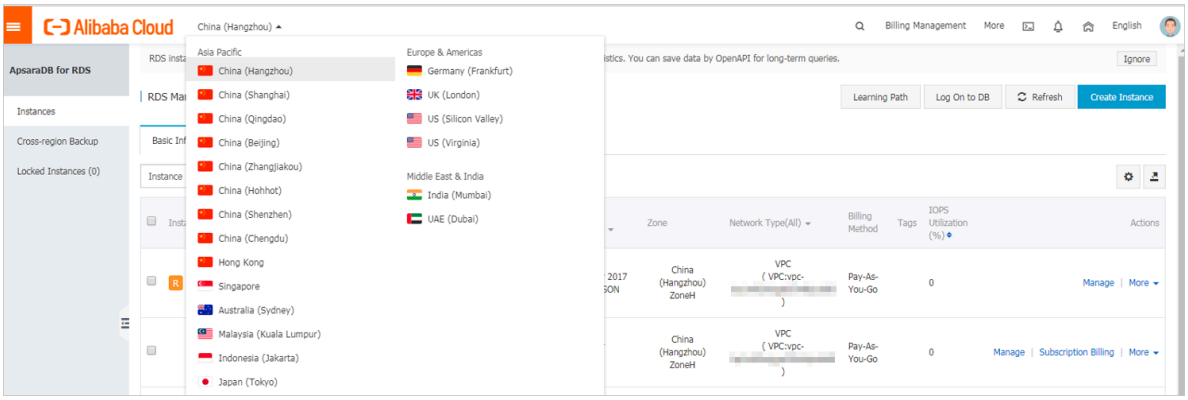
- **MySQL 8.0 High-availability Edition**
- **MySQL 5.7 High-availability Edition**
- **MySQL 5.6**

Precautions

- **The validity period of an SSL CA certificate is one year. You must renew the validity period of the SSL CA certificate in your application or client within one year. Otherwise, your application or client that uses an encrypted network connection cannot connect to RDS properly.**
- **SSL encryption increases CPU usage. Therefore, we recommend that you enable SSL encryption only for public endpoints when required. In typical cases, private endpoints do not require SSL encryption.**
- **Read/write splitting addresses do not support SSL encryption.**
- **Disabling SSL encryption restarts your RDS instance.**

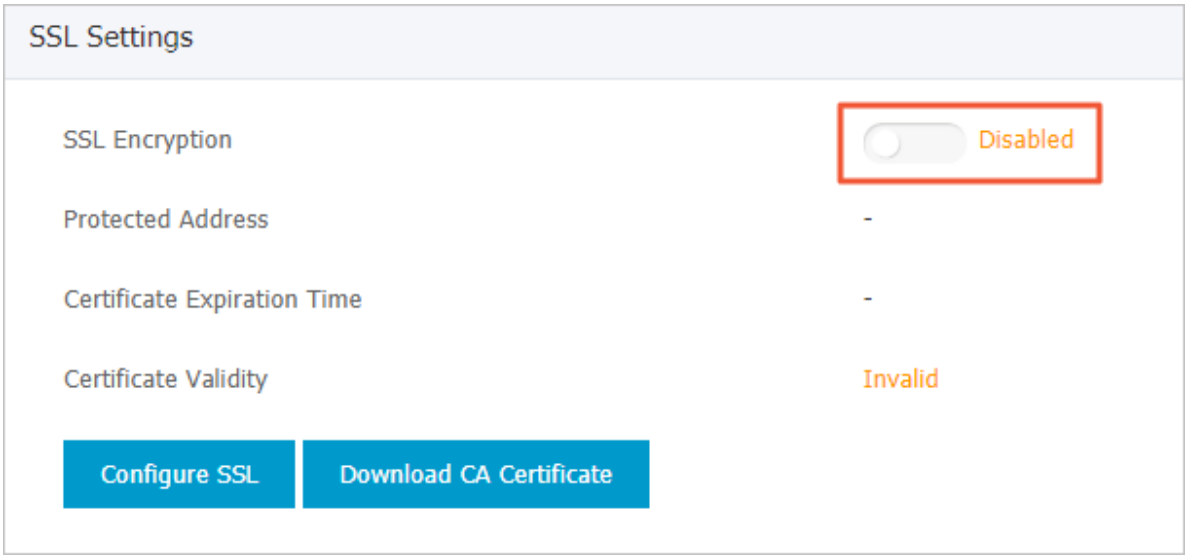
Enable SSL encryption

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**



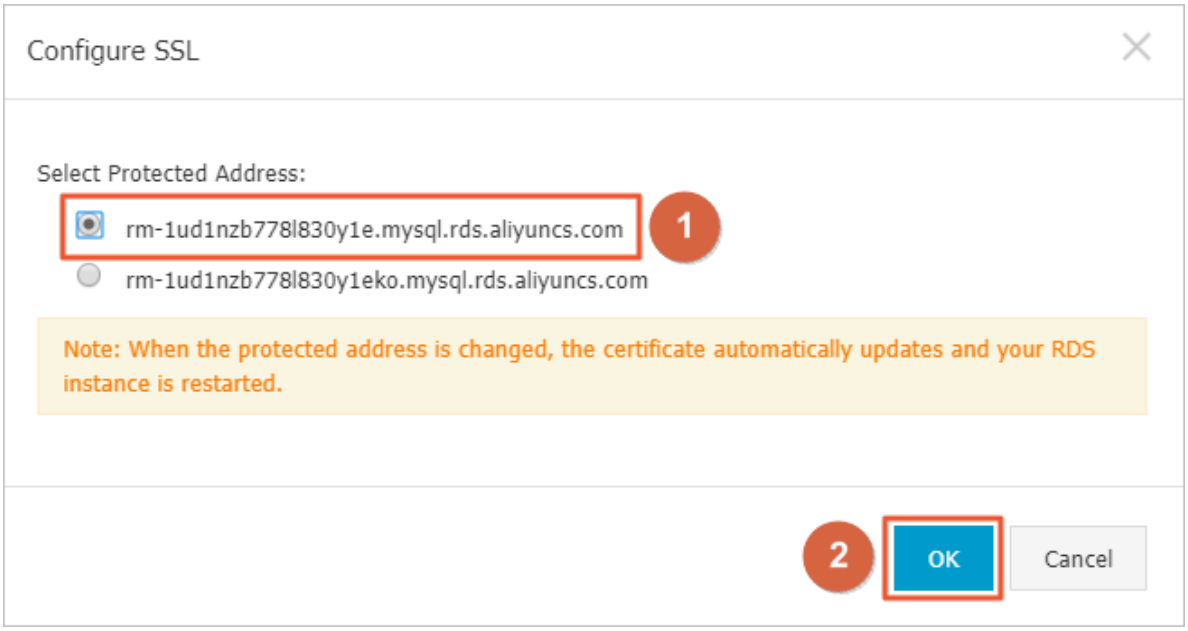
3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Data Security.**
5. **Click the SSL Encryption tab.**

6. **Click the switch next to Disabled in the SSL Encryption parameter.**

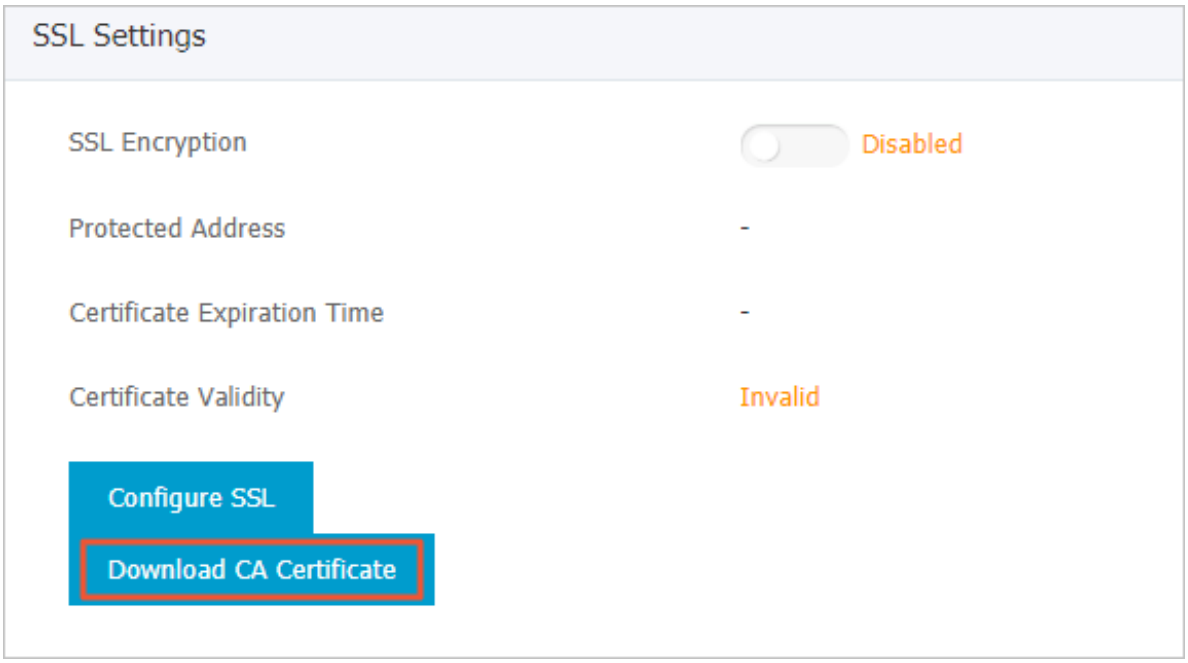


7. **In the Configure SSL dialog box, select the endpoint for which you want to enable SSL encryption, then click OK.**

Note:
You can choose to encrypt the private or public endpoint, but note that you can encrypt only one endpoint.



8. **Click Download CA Certificate to download the SSL CA certificate files in a compressed package.**



The compressed package consists of the following three files:

- **.p7b file: used to import CA certificate files in Windows operating systems.**
- **.pem file: used to import CA certificate files in other systems or applications.**
- **.jks file: used to import link CA certificate files in Java-based applications. The .jks file is stored in the TrustStore of Java.**

> **Note:**
> **When you use the .jks file in JDK 7 or JDK 8, you must modify the default JDK security configuration. Specifically, you must find the** *jre/lib/security/java.security* **file on the server where the database you want to access through SSL is located, and then reconfigure the file as follows:**

```
jdk.tls.disabledAlgorithms=SSLv3, RC4, DH keySize < 224
```

```
jdk.certpath.disabledAlgorithms=MD2, RSA keySize < 1024
```

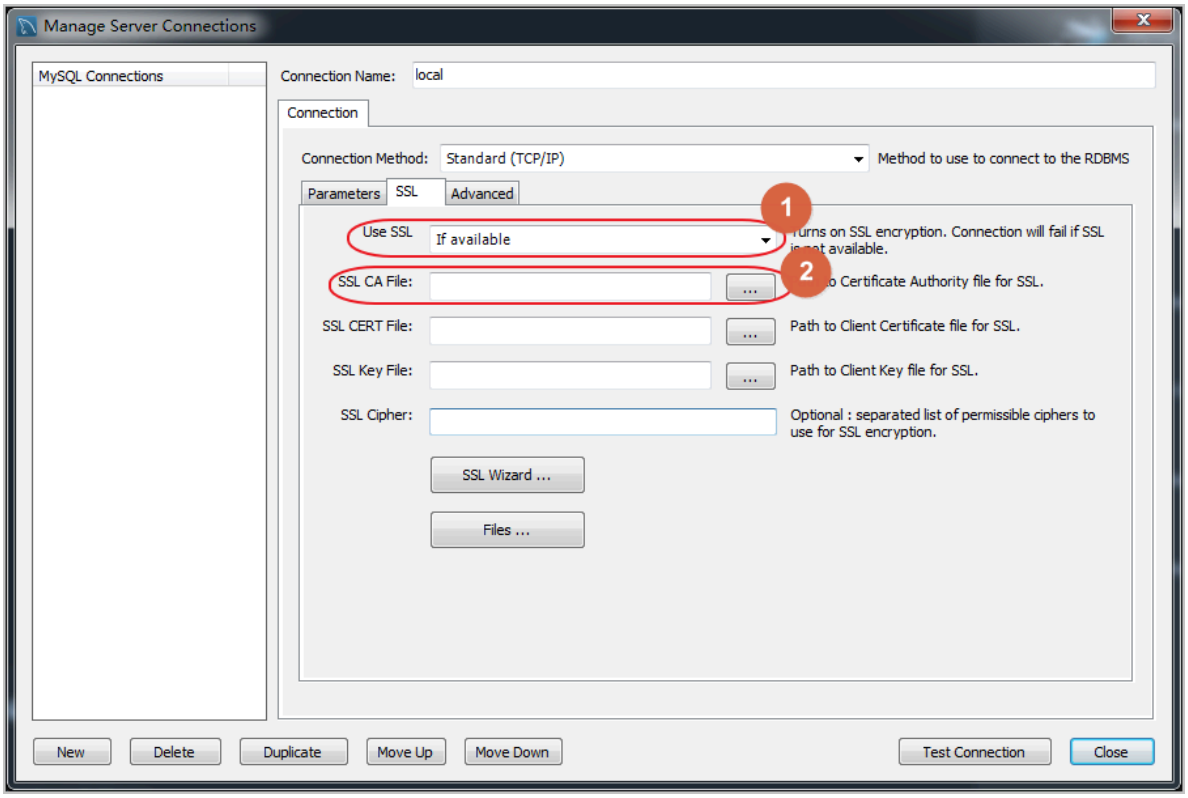
If you do not modify the JDK security configuration, the system reports errors similar to the following:

```
javax.net.ssl.SSLHandshakeException: DHPublicKey does not comply
to algorithm constraints
```

Configure the SSL CA certificate

After SSL encryption is enabled, you must configure the SSL CA certificate for your application or client when connecting to RDS. This section uses MySQL Workbench as an example to describe how to install the SSL CA certificate.

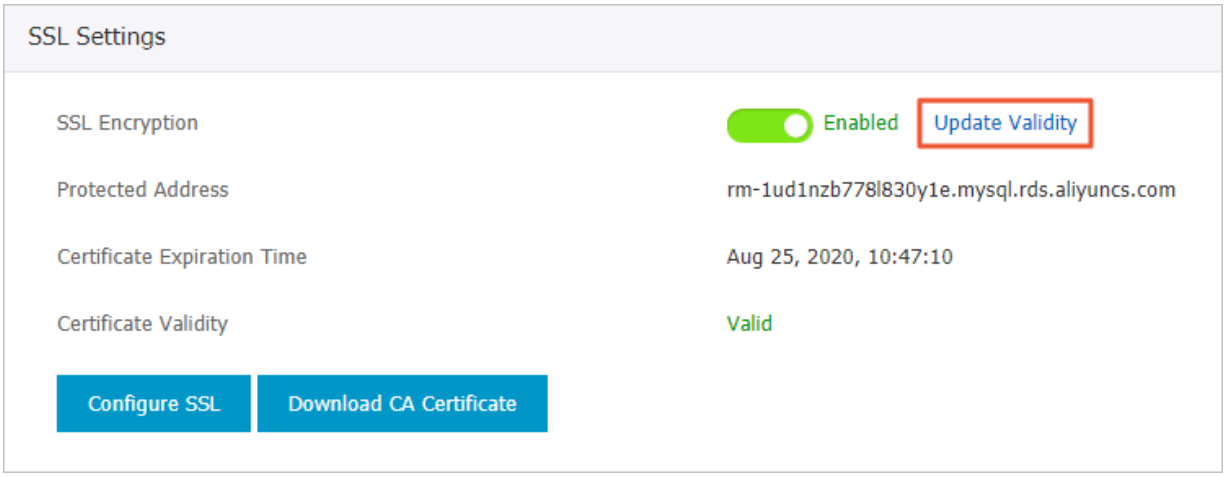
1. **Start MyQL Workbench.**
2. **Choose Database > Manage Connections.**
3. **Enable Use SSL and import the SSL CA certificate files.**



Renew the validity period of the SSL CA certificate

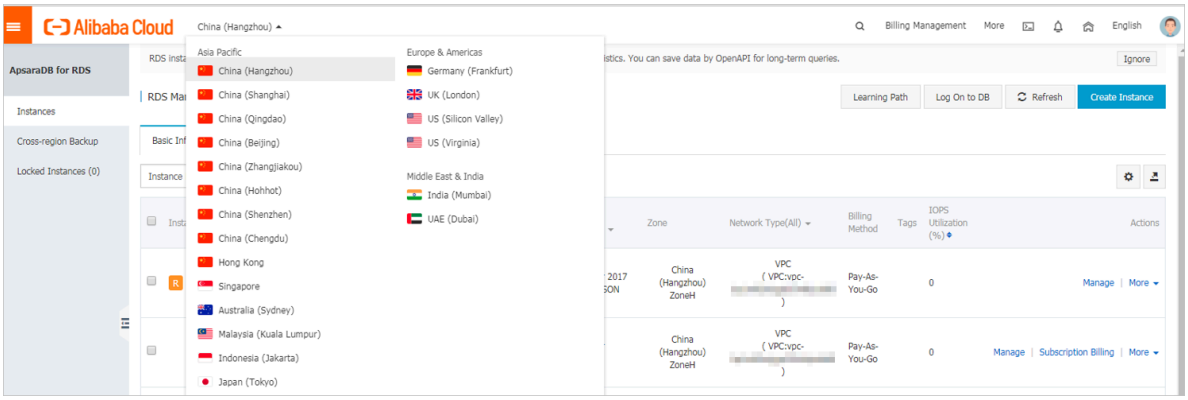
Note:

This operation causes your RDS instance to restart. You must make proper service arrangements before this operation.



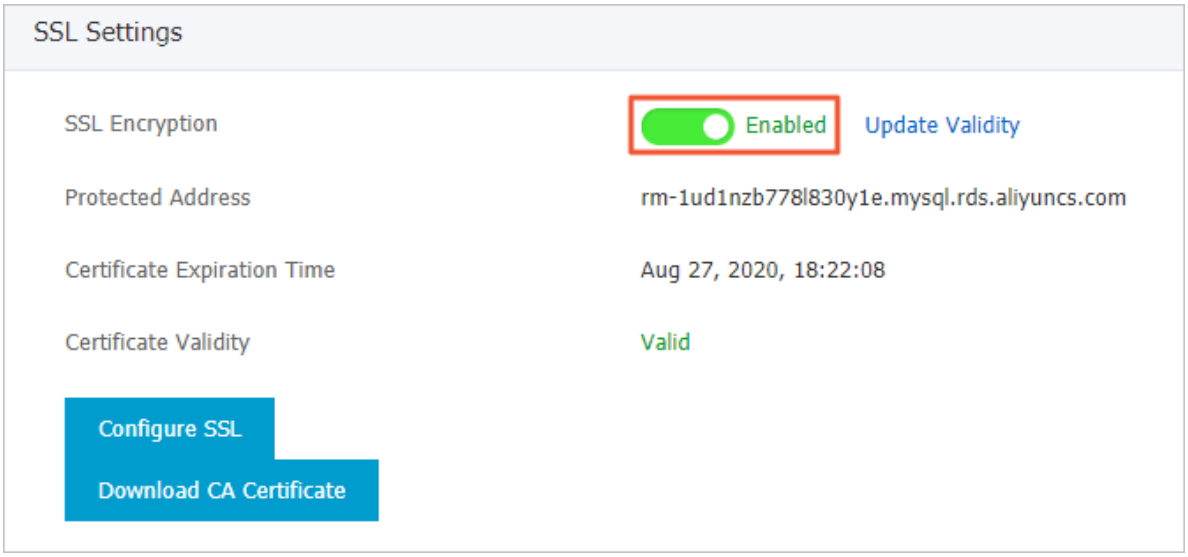
Disable SSL encryption

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**



3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Data Security.**
5. **Click the SSL Encryption tab.**

6. **Click the switch next to Enabled in the SSL Encryption parameter. In the displayed Disable SSL Encryption dialog box, click OK.**



14.4 Set TDE for an RDS for MySQL instance

This topic describes how to set Transparent Data Encryption (TDE) for an RDS for MySQL instance. With TDE enabled, RDS can encrypt and decrypt incoming and outgoing data files in real time. Specifically, RDS encrypts data before the data is written into the disk, and decrypts data when the data is read from the disk to the memory. TDE does not increase the size of data files. Developers can use the TDE function without changing any applications.

Prerequisites

- **The DB engine version of your RDS instance is MySQL 5.6.**
- **Key Management Service (KMS) has been activated. If you have not activated KMS, you can activate it as instructed when activating TDE.**

Precautions

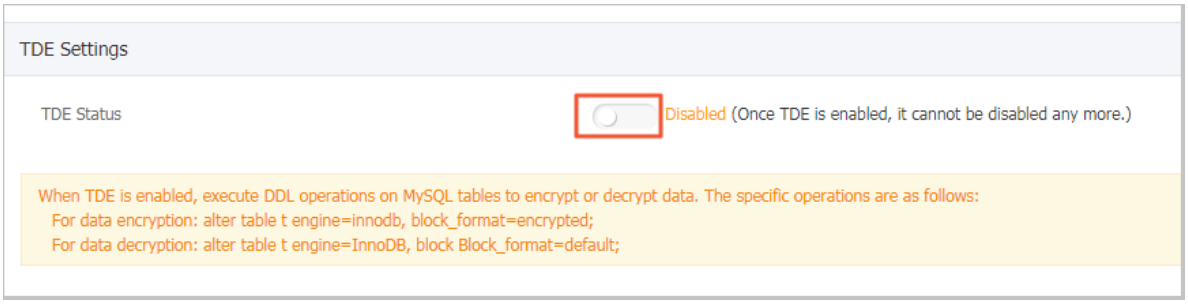
- **TDE cannot be disabled once it is activated.**
- **After TDE is activated, you cannot change keys.**
- **After TDE is activated, if you want to restore data to your computer, you must first use RDS to *decrypt data*.**
- **TDE increases CPU usage.**

Use a key automatically generated by Alibaba Cloud

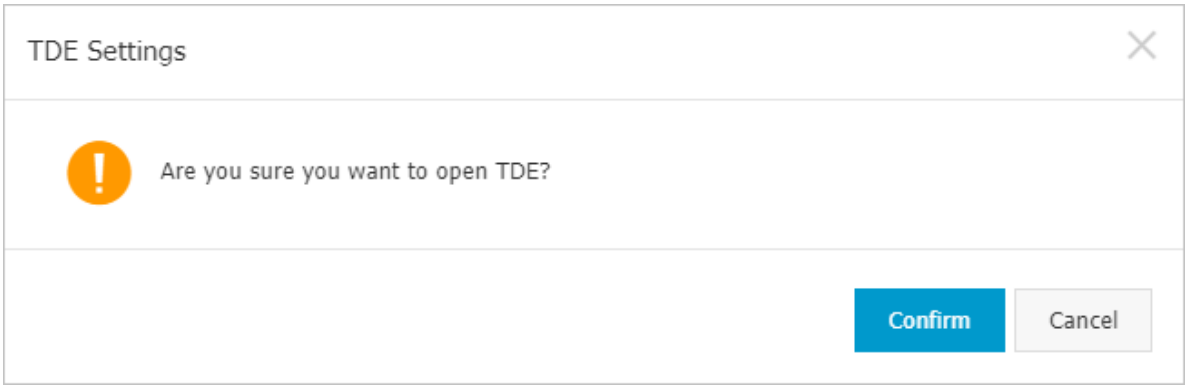
1. **Log on to the *RDS console*.**

2. In the upper-left corner, select the region where the target RDS instance is located.

3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Data Security.
5. On the TDE tab, find TDE Status and click the switch next to Disabled.



6. In the displayed dialog box, click Confirm.



Encrypt a table

Log on to the target database and run the following command to encrypt a table:

```
alter table <tablename> engine=innodb,block_format=encrypted;
```

Decrypt a table

Run the following command to decrypt a table that has been encrypted by using TDE:

```
alter table <tablename> engine=innodb,block_format=default;
```

APIs

API	Description
#unique_175	**Used to enable TDE for an RDS instance.**

15 SQL audit and historical events

15.1 SQL audit

This topic introduces the SQL audit function of RDS for MySQL.

RDS for MySQL has upgraded the SQL audit function as SQL Explorer, which continues to provide security audit and performance diagnosis but has more diverse features and costs much less. The upgrade process does not affect the RDS for MySQL instances. For more information, see *SQL Explorer*.

15.2 SQL Explorer

This topic describes how to use the SQL Explorer function. ApsaraDB RDS for MySQL has upgraded the SQL audit function as SQL Explorer, which continues to provide security audit and performance diagnosis but has more diverse features and costs much less. The upgrade process does not affect the RDS for MySQL instances.

Prerequisites

The SQL Explorer function is available only to the following DB engine versions and editions:

- **MySQL 8.0 High-availability Edition (with local SSDs)**
- **MySQL 5.7 High-availability Edition**
- **MySQL 5.6**
- **MySQL 5.5**

Context

With the SQL Explorer function enabled, the system records information about all DML and DDL operations. The information is obtained through network protocol analysis and consumes only a small amount of CPU resources. The Trial Edition can retain SQL log data of one day free of charge. If you want to retain your SQL log data for more than one day, you must pay for storage.

Scenarios

The SQL Explorer function is suitable to the following scenarios:

- **The enterprise has high requirements on data security. Such enterprises come from the finance, security, stocks, governmental affairs, and insurance sectors.**
- **The database running needs to be analyzed to locate problems or verify SQL statement performance.**
- **The data needs to be restored by using the SQL statements in SQL log data.**

Differences between SQL log and binary log

You can view incremental data through SQL log or binary log. However, these two types of logs are different:

- **SQL log: similar to the auditing log of MySQL and records information about all DML and DDL operations. The information is obtained through network protocol analysis. SQL log does not parse the actual parameter values, and therefore a small amount of data may be lost if a large amount of SQL query statements are executed. If you collect statistics on incremental data by using SQL log, the obtained incremental data may be lost.**
- **Binary log: records information about all add, delete, and modify operations and the incremental data used for data restoration. Binary log data is temporarily stored in the RDS instance. The system periodically transfers full binary log files to OSS. OSS then stores the files for seven days. However, the binary log files to which data is being written cannot be saved. Therefore, some binary log files may fail to be uploaded to OSS. Binary log accurately records the incremental data but cannot help you obtain real-time log data.**

Precautions

The time range for online query extends up to 24 hours. This is because SQL Explorer records all database-related operations, which involve a large amount of SQL statements. If the time range exceeds 24 hours, the query takes a long time and may even time out.

Note:

If you want to query SQL log data within a time range longer than 24 hours, we recommend that you export the SQL log data to your computer and then query the needed data.

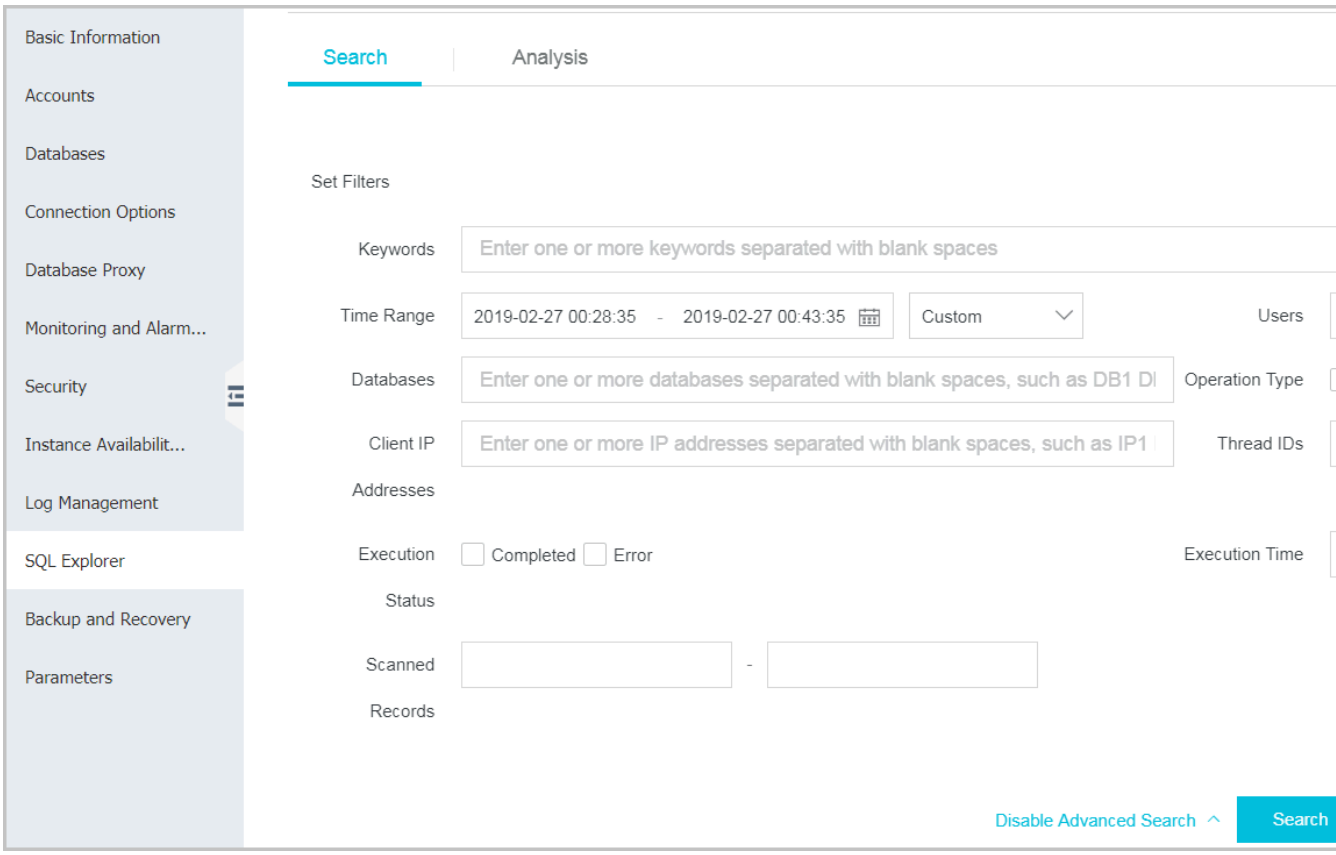
Features

- **SQL log**

  SQL log records all operations that have been performed on databases. With SQL log, you can do database troubleshooting, action analysis, and security audit.

- **Enhanced search**

  You can search data by database, user, client ID, thread ID, execution time, or the number of scanned rows. You can also export and download the search results.



- **SQL analysis**

  This new feature provides visualized interactive analysis of SQL log of a specified time period. You can use this feature to locate abnormal SQL statements and performance issues.

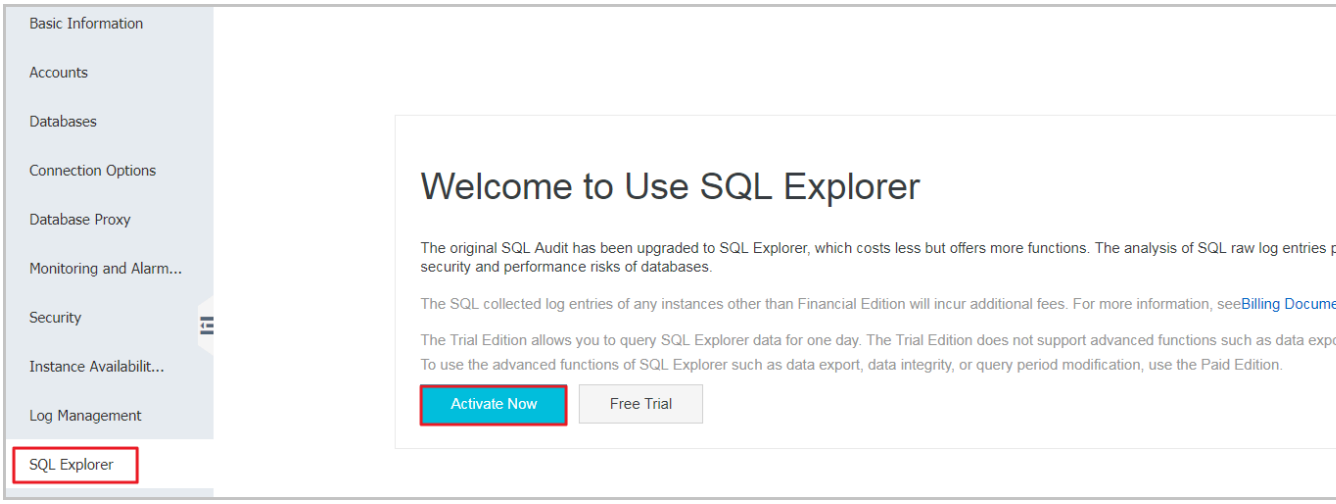
- **Cost reduction**

  SQL explorer adopts the column-based storage and compression technology to reduce the SQL log size and reduce storage costs by about 60%. The hourly fee is US$ 0.0018 per GB.

Activate SQL Explorer

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**

3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click SQL Explorer.**
5. **Click Activate Now.**



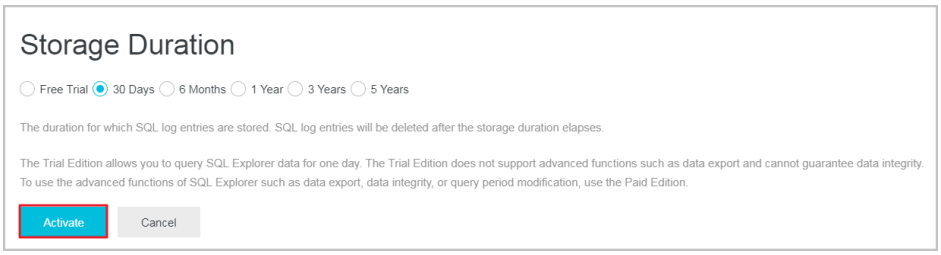
6. **Specify the SQL log storage duration (for how long you want to keep the SQL log), and click Activate.**

> **Note:**
> **When the storage duration of an SQL log file exceeds the specified duration, the file is deleted.**

- **Trial Edition: You can use SQL Explorer for a long time. However, the SQL log files are retained for only one day, which means that you can query only the**

SQL log data of one day. Additionally, you cannot use advanced functions such as data export, and data integrity cannot be guaranteed.

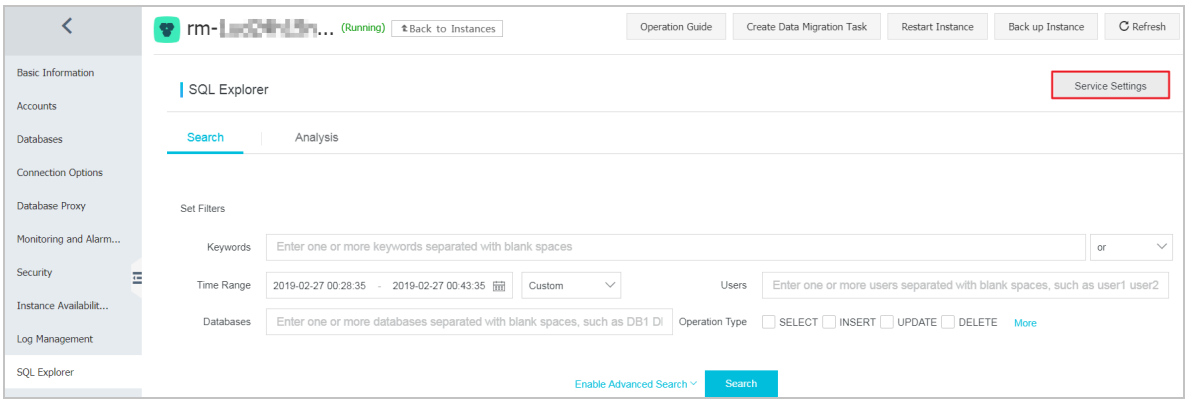
- Non-Trial Edition: The SQL log files can be stored for 30 days, six months, one year, three years, or five years. You are charged an hourly fee of USD 0.0018 per GB.



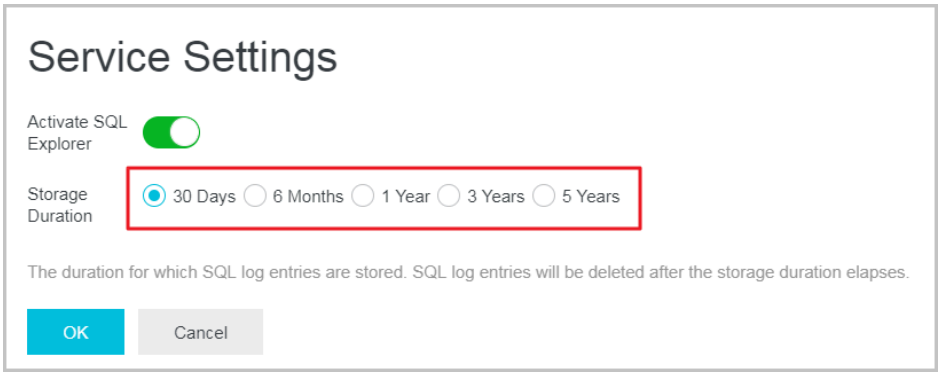
Change the SQL log storage duration

1. Log on to the *RDS console*
2. In the upper-left corner, select the region where the target RDS instance is located.

3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click SQL Explorer.
5. Click Service Settings.



6. **Change the storage duration as needed.**



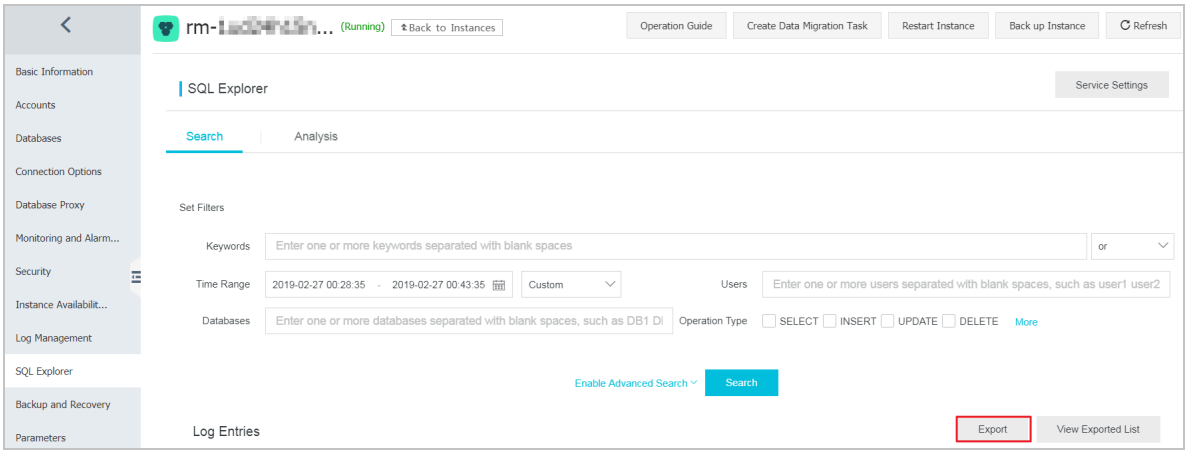
Disable SQL Explorer

Note:
If you disable the SQL Explorer function, the existing SQL log files are deleted. We recommend that you export and save the SQL log file to your computers before you disable the function.

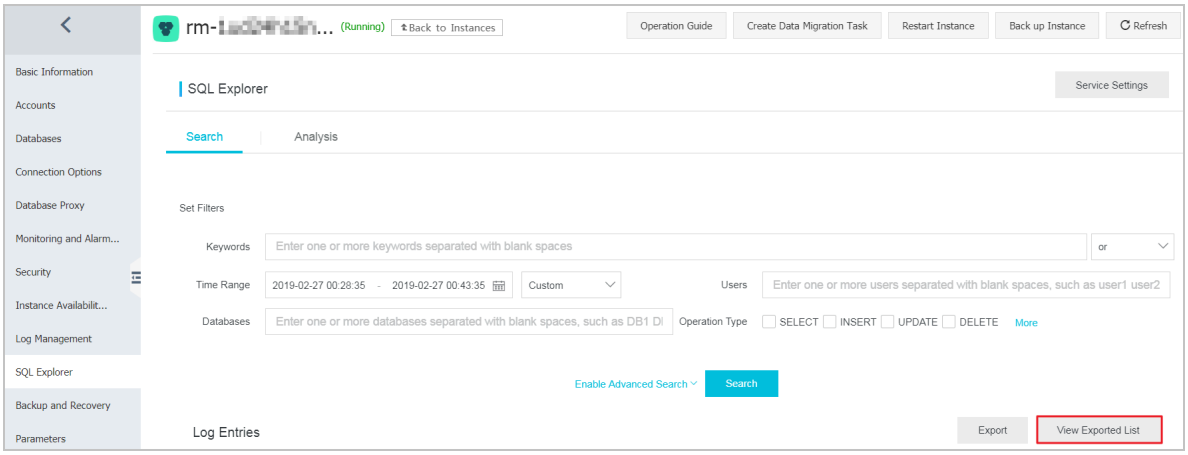
1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**

3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click SQL Explorer.**

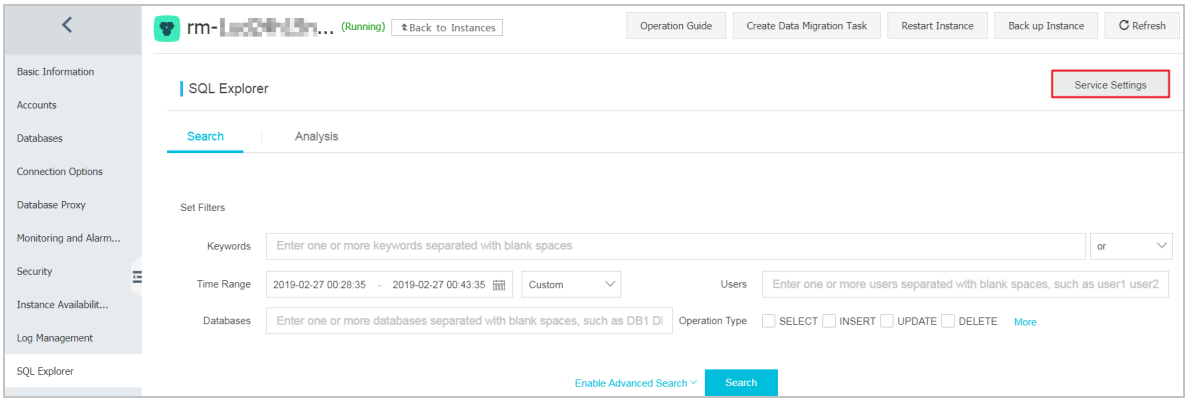
5. Click Export.



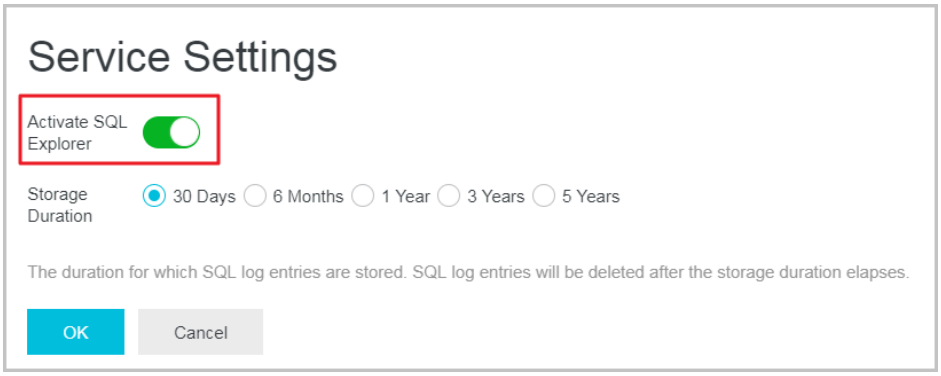
6. In the displayed dialog box, click OK.
7. After the export is complete, click View Exported List and download the log file to your computer.



8. Click Service Settings.



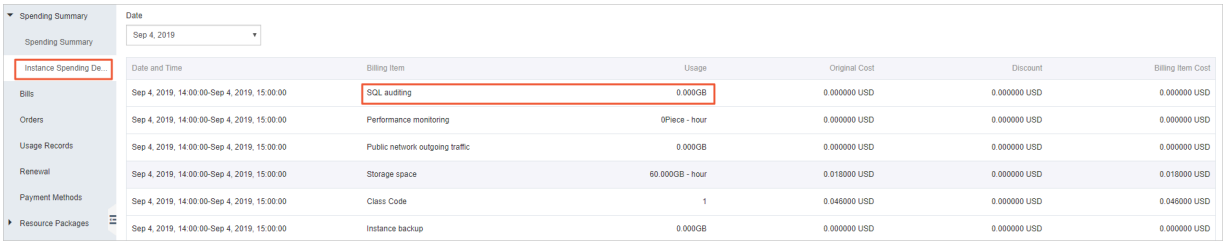
9. **Click the switch next to Activate SQL Explorer to disable SQL Explorer.**



FAQ

How do I obtain the SQL log size after enabling SQL Explorer?

Log on to the RDS console. In the upper-right corner, choose Billing Management > Billing Management. In the left-side navigation pane, choose Spending Summary > Instance Spending Detail. Then you can find the SQL log size for the target RDS instance.



Note:

SQL auditing indicates the SQL log size.

16 Database backup

16.1 Back up the data of an RDS for MySQL instance

Backup data is used to cope with data loss or damage. You can set a backup policy for the system to automatically back up the data and logs of an RDS for MySQL instance. Alternatively, you can manually back up the data and logs.

> **Note:**
> **This topic describes the default backup function, which stores backup files to the region where your RDS instance is located. You can also store the backup files to another region. For more information, see** *Back up data across regions*.

Fees

Each RDS instance has a specific quota of free backup space. If the size of backup files exceeds the quota, additional fees are incurred. We recommend that you specify a proper backup cycle that can meet your service requirements while also maximizing backup space utilization. For more information about the quota of free backup space, see *View the quota of free backup space for an RDS for MySQL instance*.

> **Note:**
> - **For more information about pricing and billable items, see** *#unique_20*.
> - **For more information about the metering standards on backup space usage, see** *ApsaraDB RDS for MySQL pricing*.

Precautions

- **Do not perform DDL operations during the backup. Otherwise, tables are locked and consequently the backup fails.**
- **Back up data and logs during off-peak hours.**
- **If the data volume is large, the backup may take a long time.**
- **Backup files are retained for a specified time period. Download the backup files to your computer before they are deleted.**

Overview

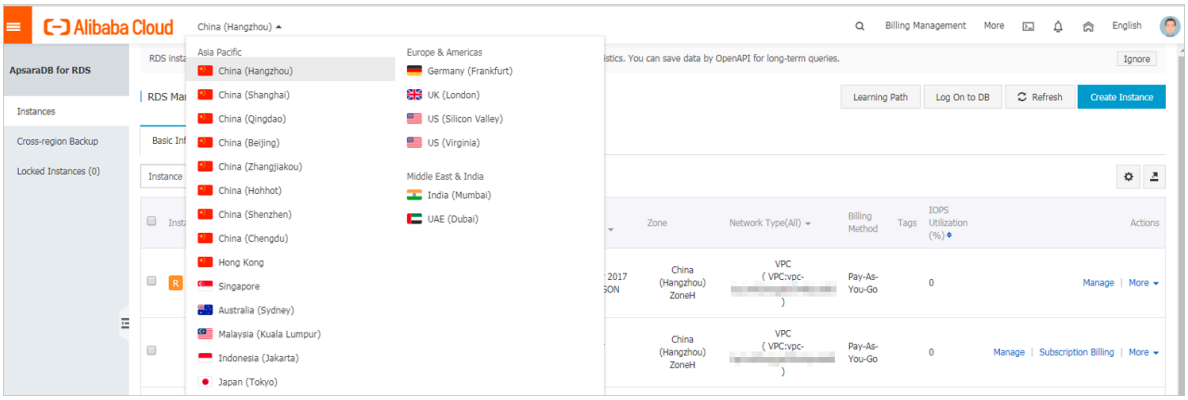
Data backup	Log backup
· MySQL 5.5/5.6/5.7/8.0 High-availability/Enterprise Edition (with local SSDs): - Automatic backup supports full physical backup. - Manual backup supports full physical backup, full logical backup, and single-database logical backup. · MySQL 5.7/8.0 High-availability Edition (with SSDs or ESSDs): Only snapshot backup is supported. You can restore data from a snapshot to a new RDS instance but cannot download the data. · MySQL 5.7/8.0 Basic Edition (with SSDs): Only snapshot backup is supported. You can restore data from a snapshot to a new RDS instance but cannot download the data.	· Binary log files occupy the disk space of an RDS instance. · If the size of a binary log file exceeds 500 MB or data has been being written into a binary log file for more than 6 hours, the system stops writing data into this file and starts to write data into another file. The earlier binary log file is then uploaded asynchronously. · You can upload binary log files to OSS with a few clicks without interrupting the data restoration function. Additionally, binary log files do not occupy disk space. **Note:** · Basic Edition does not allow you to upload binary log files with a few clicks. · You cannot access the OSS bucket where binary log files are located.

Set a backup policy for automatic backup

ApsaraDB for RDS can automatically back up databases according to the backup policy you set.

1. **Log on to the *RDS console*.**

2. In the upper-left corner, select the region where the target RDS instance is located.

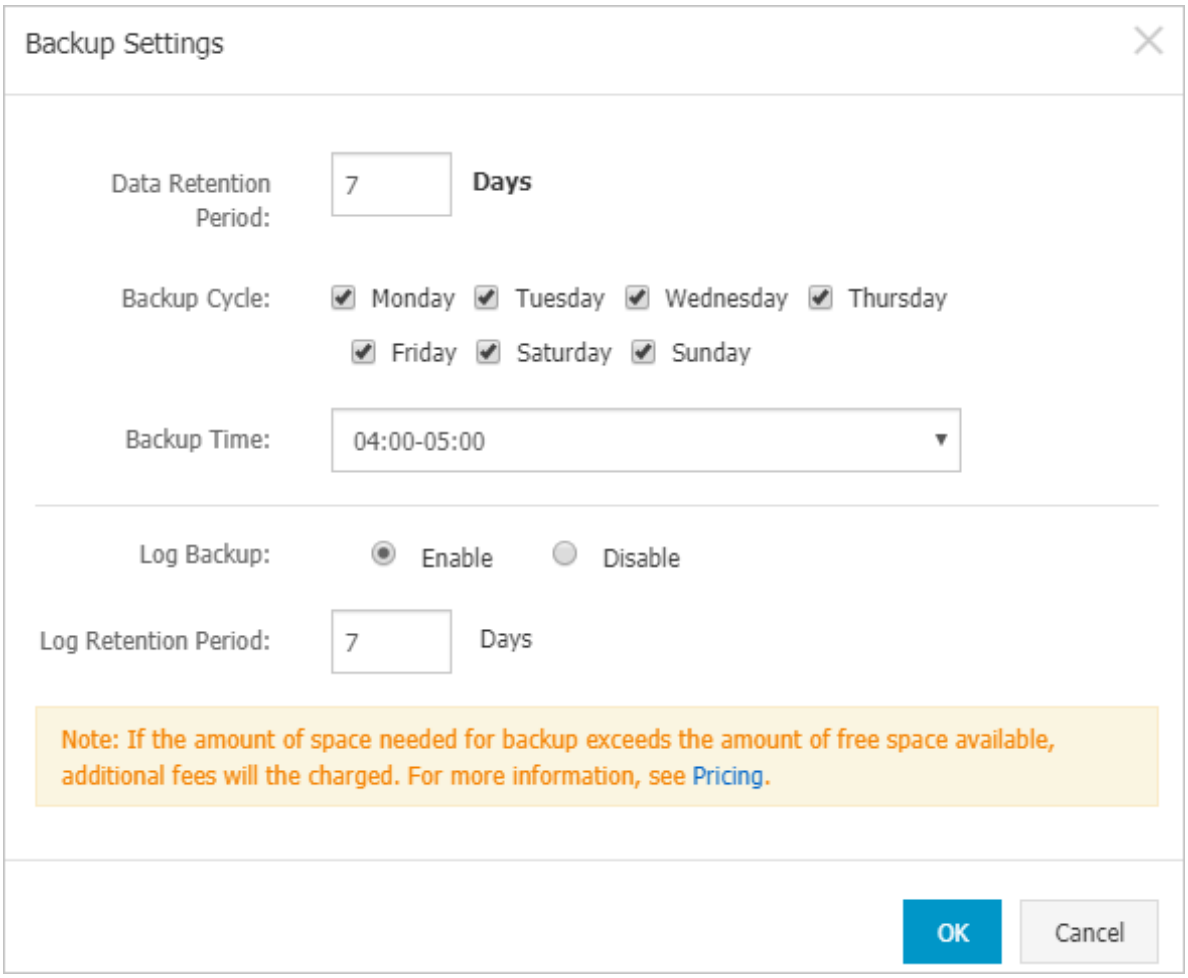


3. Find the target RDS instance and click its ID.
4. In the left-side navigation pane, click Backup and Restoration.
5. On the Backup and Restoration page, click the Backup Settings tab. On the Backup Settings tab, click Edit.
6. In the Backup Settings dialog box, set the backup parameters and click OK. The following table describes the parameters.

Table 16-1: Backup parameters

Parameter	Description
Data Retention Period	The data retention period spans from 7 days to 730 days. The default retention period is 7 days. **Note:** For MySQL 5.7 Basic Edition (with SSDs), the data retention period is 7 days and cannot be changed.
Backup Cycle	Select one or more workdays.
Backup Time	You can select any time period, which is measured in the unit of hour. We recommend that you select a time period during off-peak hours.
Log Backup	The status of the log backup function. **Notice:** If you disable the log backup function, all log backup files are deleted and the time-based data restoration function becomes unavailable.

Parameter	Description
Log Retention Period	· The number of days in which log backup files are retained. The default retention period is 7 days. · The log retention period spans from 7 days to 730 days and must be shorter than or equal to the data retention period. **Note:** For MySQL 5.7 Basic Edition (with SSDs), the log retention period is 7 days and cannot be changed.

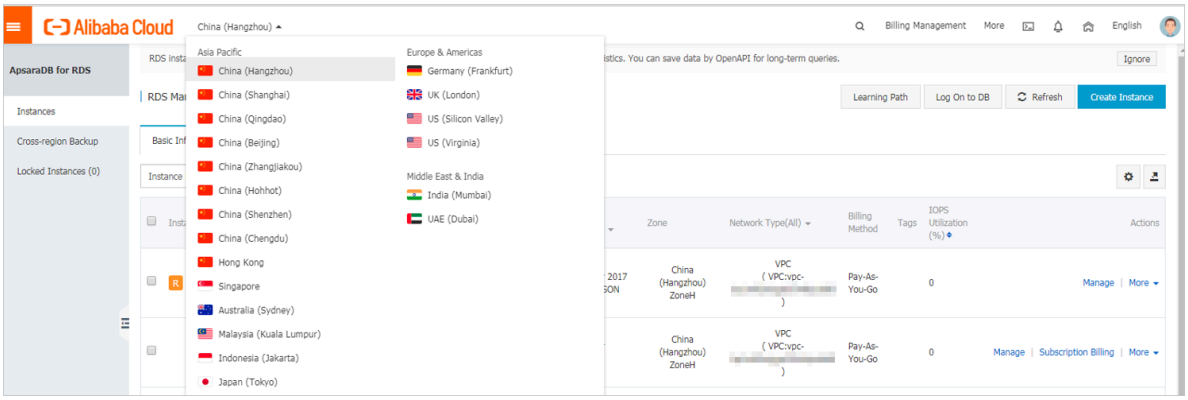


Manually back up data

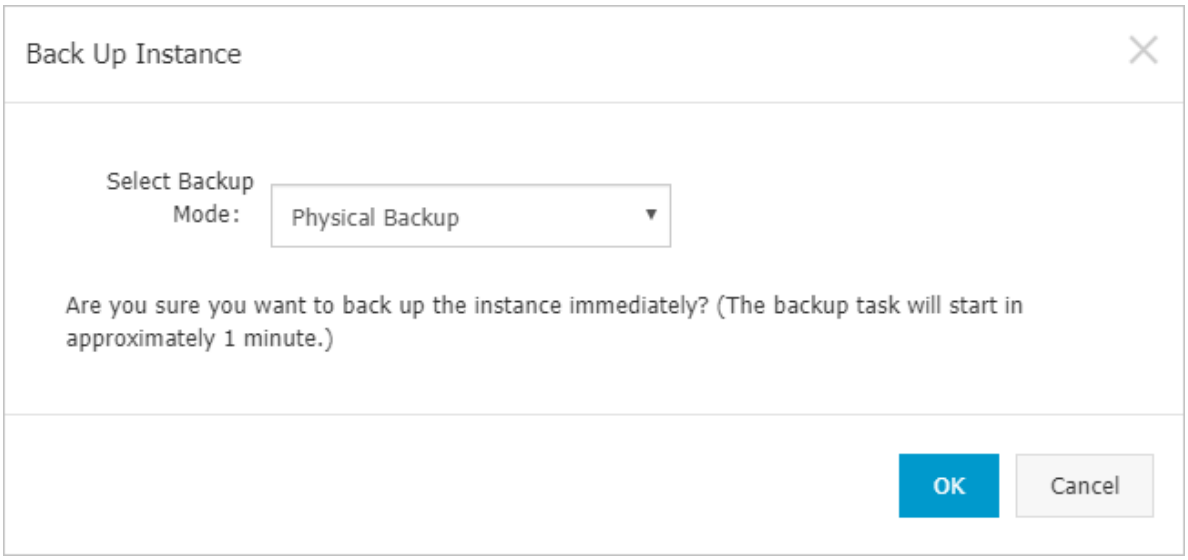
This section uses the single-database logical backup for MySQL 5.7 High-availability Edition (with local SSDs) as an example.

1. **Log on to the *RDS console*.**

2. In the upper-left corner, select the region where the target RDS instance is located.



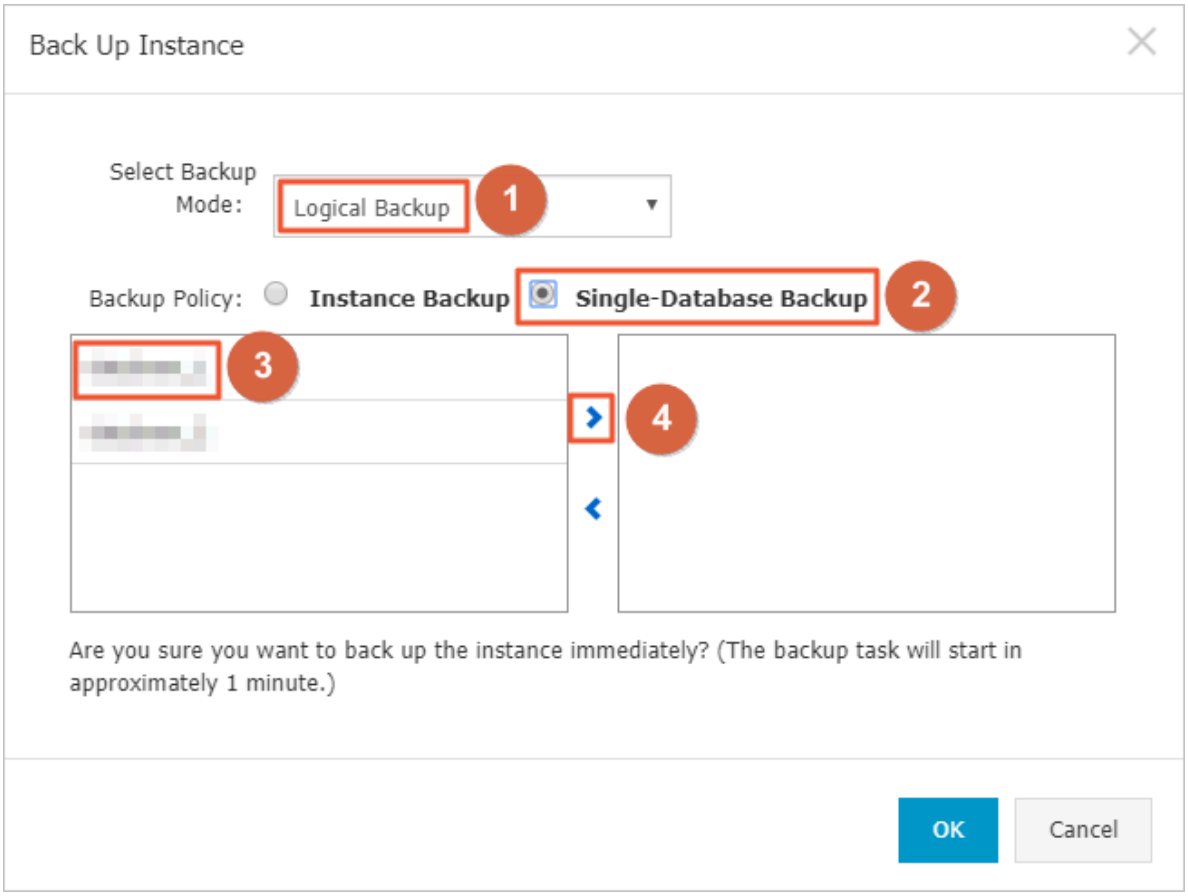
3. Find the target RDS instance and click its ID.
4. In the upper-right corner of the Basic Information page, click Back Up Instance.
5. In the Back Up Instance dialog box, set the backup parameters and click OK.



Note:

If you set the Select Backup Mode parameter to Logical Backup and the Backup Policy parameter to Single-Database Backup, select a database from the left and click > to add the database to the list on the right. If the RDS instance does

not have a database, you must create databases according to *Create accounts and databases for an RDS for MySQL instance*.



FAQ

1. **Can I disable the data backup function for an RDS for MySQL instance?**

   No. The data backup function must be enabled. However, you can lower the backup frequency to at least two times a week.

2. **Can I disable the log backup function for an RDS for MySQL instance?**

   Yes. You can disable the log backup function if your RDS instance does not run the MySQL Basic Edition.

3. **Why does the backup fail sometimes?**

   If you perform DDL operations during the backup, tables are locked and consequently the backup fails.

Reference

- *Download the data backup files and log backup files of an RDS for MySQL instance*
- *Restore the data of an RDS for MySQL instance*

APIs

API	Description
#unique_183	**Used to create a backup set.**
#unique_184	**Used to list backup sets.**

16.2 View the quota of free backup space for an RDS for MySQL instance

This topic describes how to calculate and view the quota of free backup space for an RDS for MySQL instance. The quota varies depending on the used DB engine version and edition. Additionally, this topic describes how to calculate the backup space beyond the quota.

Backup files occupy backup space. Each RDS instance has a specific quota of free backup space. If the total size of backup files exceeds the quota, additional fees are incurred.

Calculate the quota of free backup space and the backup space beyond the quota

Quota of free backup space = Round up (50% × Storage space purchased for the RDS instance) (Unit: GB)

Backup space beyond the quota = Backup data size + Backup log size - Round up (50 % × Storage space purchased for the RDS instance) (Unit: GB)

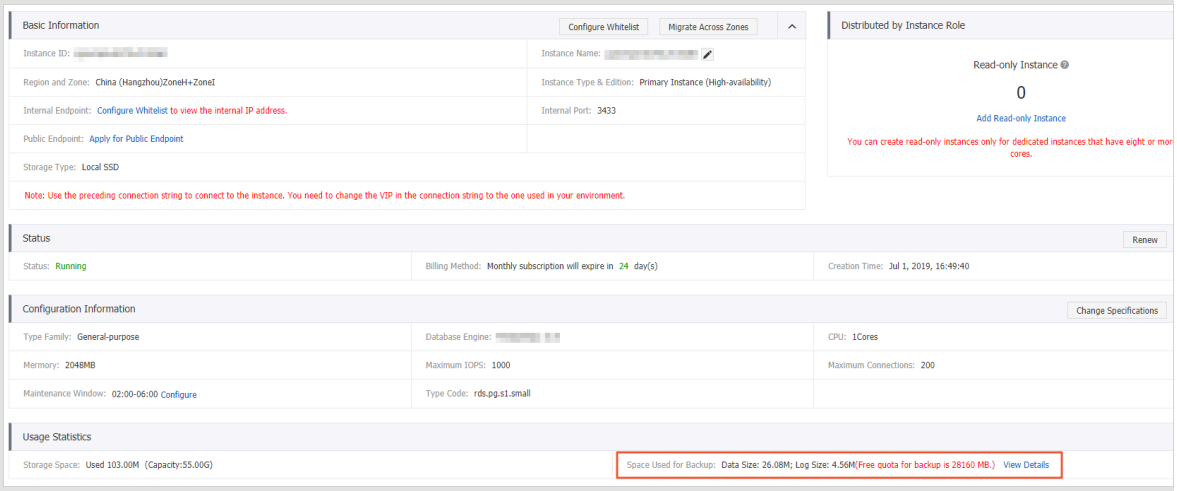
For example, the backup data size is 30 GB, the backup log size is 10 GB, and the storage space is 60 GB, then you must pay for 10-GB storage space every hour:

```
Hourly fees = 30 + 10 - 50% × 60 = 10 (GB)
```

Note:

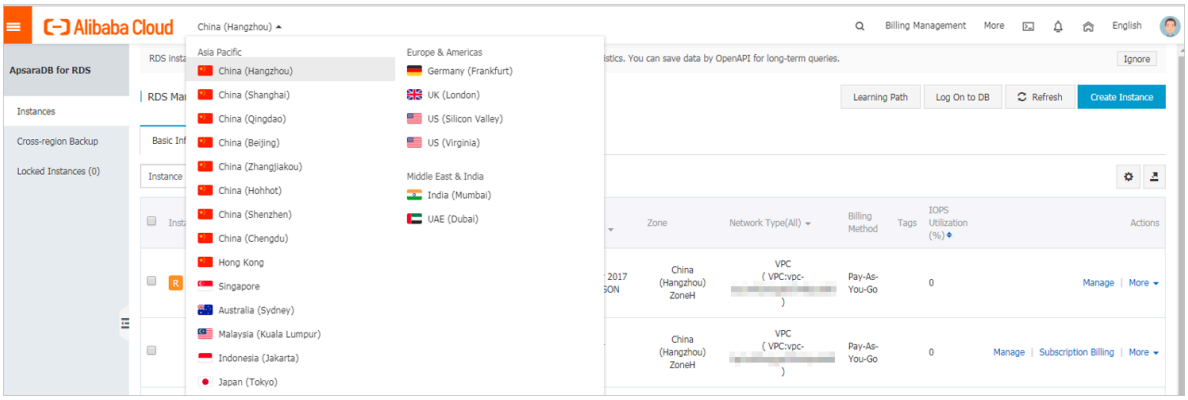
- **For more information about the hourly fees for the backup space beyond the quota, see** *ApsaraDB RDS for MySQL pricing*.

- **The Basic Editions of some DB engines store backup files generated within the last seven days for free. For more information, log on to the RDS console.**



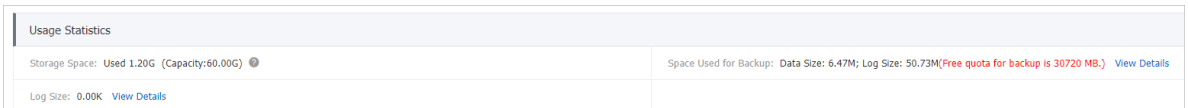
View the quota of free backup space in the RDS console

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**



3. **Find the target RDS instance and click its ID.**
4. **In the Usage Statistics section of the Basic Information page, view the data size next to Space Used for Backup. The data size is the quota of free backup space.**

> **Note:**
> **The quota of free backup space varies depending on the instance type. The following figure is only an example.**



16.3 Download the data backup files and log backup files of an RDS for MySQL instance

This topic describes how to download the data backup files and log backup files of an RDS for MySQL instance. The downloaded log backup files are not encrypted. You can save the backup files for archiving or use them to restore the instance to an on-premises database.

> **Note:**
>
> **An RDS for MySQL instance with SSDs does not support the download of data backup files. You can** *restore the data of an RDS for MySQL instance* **or** *migrate data from RDS for MySQL to an on-premises MySQL database*.

Limits

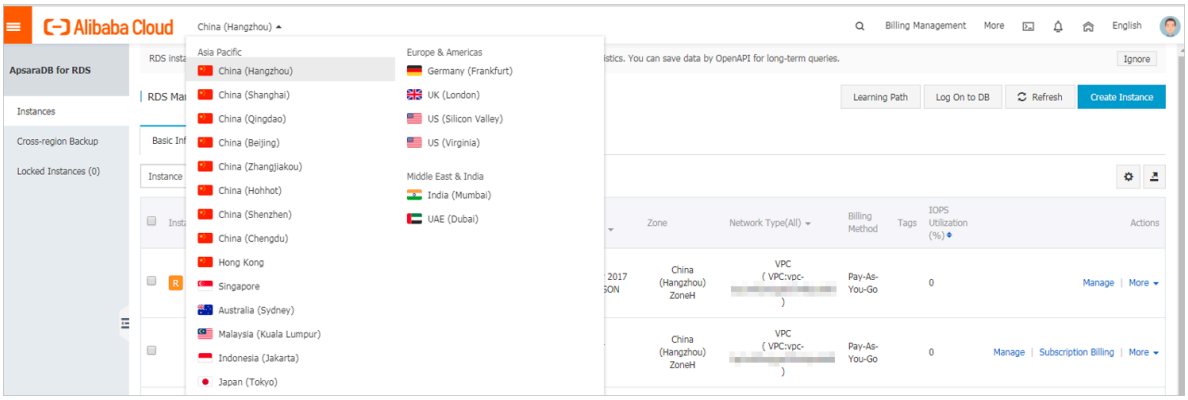
A RAM user who has only the read-only permissions cannot download backup files. You can add the required permissions to a RAM user in the RAM console. For more information, see *Grant backup file download permissions to a RAM user with only read-only permissions*.

DB engine	Data backup download	Log backup download
MySQL	· MySQL 5.5/5.6/5.7/8.0 (with local SSDs): support the download of full physical data backup files and logical data backup files. · MySQL 5.7/8.0 Basic/High-availability Edition (with ESSDs /SSDs): do not support the download of data backup files.	All versions support the download of log backup files. **Note:** For more information about binary logs, see #unique_186

Procedure

1. **Log on to the** *RDS console*.

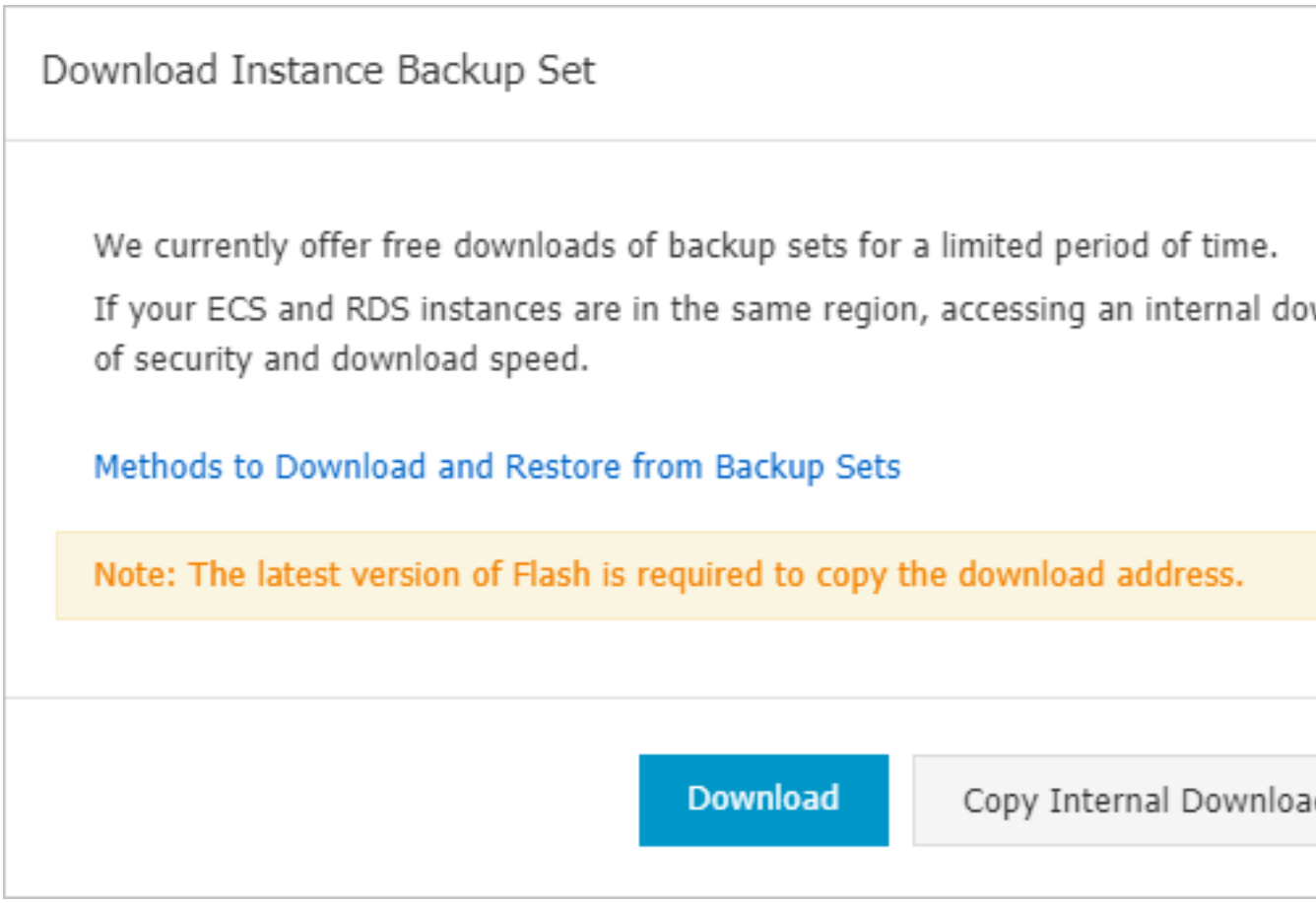
2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Backup and Restoration.
5. Click the Data Backup or Log Backup tab.
   - If you want to download data backup files, click Data Backup.
   - If you want to download log backup files, click Log Backup
6. Select a time range and click Search.
7. Find the data or log backup file you want to download, and in the Actions column click Download.

> **Note:**
>
> - If the Download button is unavailable, see *Limits*.
> - If you want to use the downloaded data backup file for data restoration, we recommend that you select the data backup file that was generated at the time point closest to the time point from which you want to restore data.
> - If you want to use the downloaded log backup file to restore your RDS instance to an on-premises database, note the following:
>   - The Instance ID of the log backup file must be the same as the Instance No. of the selected data backup file.
>   - The start time of the log backup file must be later than the generation time of the selected data backup file and earlier than the time point from which you want to restore data.

8. **In the Download Instance Backup Set or Download Binary Log dialog box, select a download method.**



Download method	Description
Download	To download the backup file through the public connection address.
Copy Internal Download URL	To copy the internal download URL only. When your ECS instance is located in the same region as the RDS instance, you can log on to your ECS instance and then use the internal download URL to download the backup file. This is faster and more secure.
Copy External Download URL	To copy the external download URL only. This method is suitable when you download the backup file by using other tools.

Note:

In a Linux operating system, you can run the following command to download a data backup file:

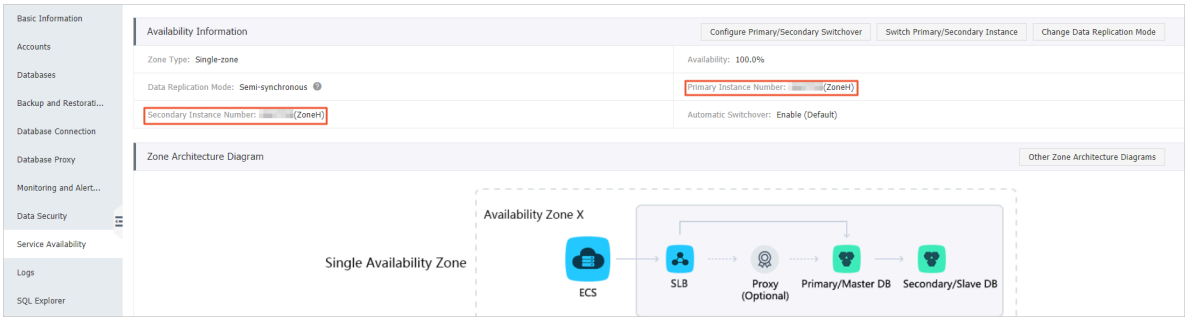
```
wget -c '<Download URL of the data backup file>' -O <User-defined
file name>.tar.gz
```

- **The `-c` parameter is used to enable resumable download.**
- **The `-O` parameter is used to save the downloaded result as a file with the specified name (the file extension is .tar.gz or .xb.gz as included in the URL).**
- **If you enter more than one download URL, then you must include each download URL in a pair of single quotation marks ('). Otherwise, the download fails.**

FAQ

1. **Why are there two binary log files that have the same name for an RDS for MySQL instance?**

   RDS for MySQL works in the master/slave HA architecture. Therefore, both the master and slave instances generate binary logs. You can distinguish between two binary log files that have the same name based on their Instance IDs. To view the IDs of the master and slave instances, you can go to the Service Availability page.



2. **What can I use the downloaded data and log backup files for?**

   You can use the downloaded data and log backup files to restore data at anytime. For more information, see *#unique_187*.

16.4 Configure local binary log policies for an RDS for MySQL instance

This topic describes how to configure local binary log policies for an RDS for MySQL instance. RDS for MySQL deletes local binary log files according to the

polices you specify. Before you set the policies, you must learn about the rules for MySQL to generate and delete binary log files.

Context

For information about the related API, see *#unique_189*.

The rules for an RDS for MySQL instance to generate binary log files are as follows:

- In typical cases, when the size of the current binary log file exceeds 500 MB, the system starts to write data into a new binary log file. When the system is writing data into the new file, the old file is uploaded asynchronously.
- In certain circumstances, the system stops writing data into the current binary log file due to reasons such as command execution and system restart even when the file size does not exceed 500 MB.
- In certain circumstances, the size of a binary log file exceeds 500 MB when, for example, the system is executing a large transaction.

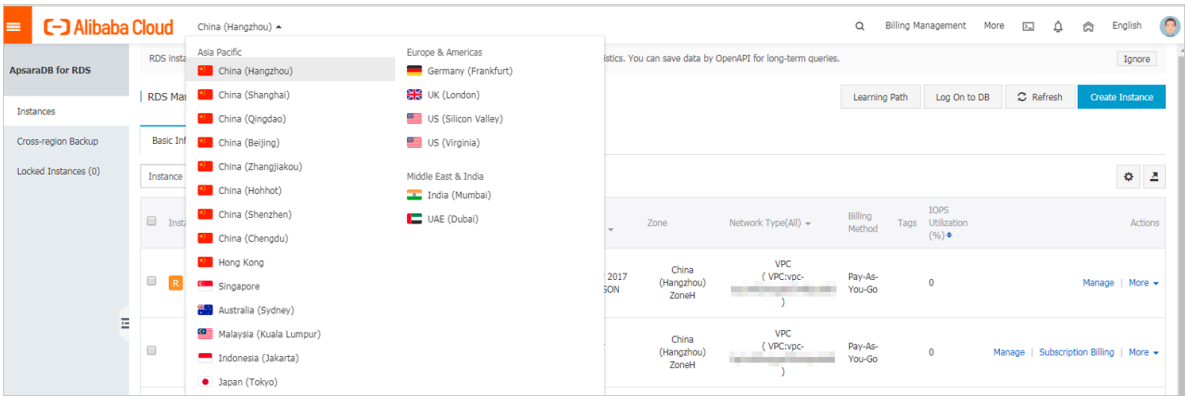
The default rules for an RDS for MySQL instance to delete binary log files are as follows:

- By default, the system retains the binary log files generated within the last 18 hours.
- When the space usage does not exceed 80%, the system retains the binary log files that occupy 30% of the space regardless of whether these files have been uploaded to OSS or not.
- When the space usage exceeds 80%, the system initiates requests for deleting local binary log files after these files are uploaded to OSS. The requests must be scheduled, and therefore these files are deleted at a certain delay.

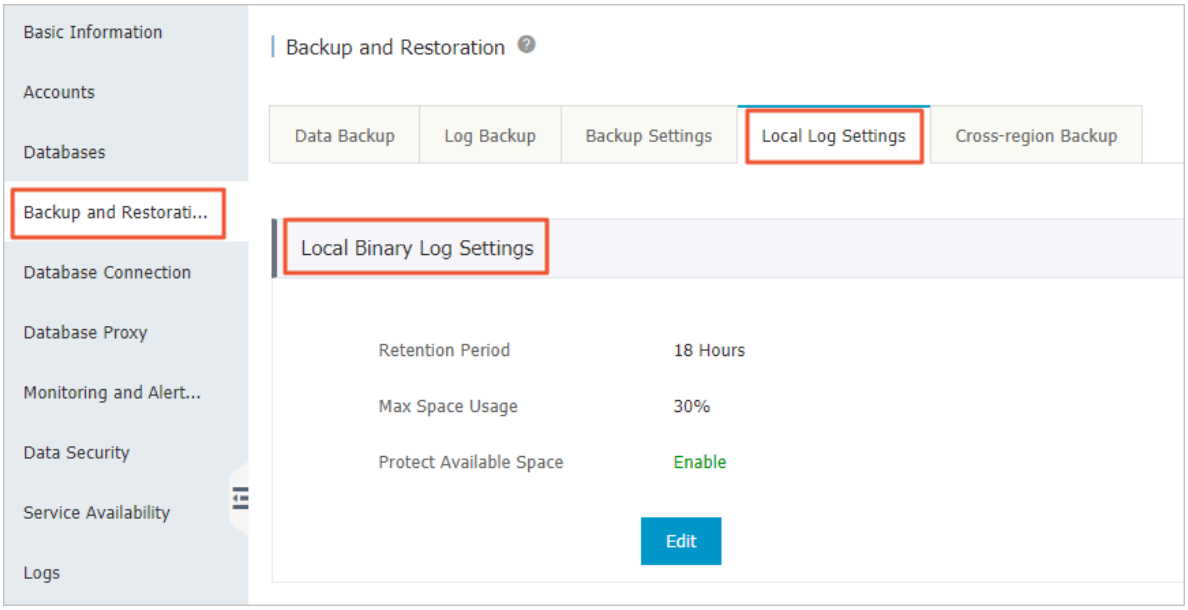
Procedure

1. Log on to the *RDS console*.

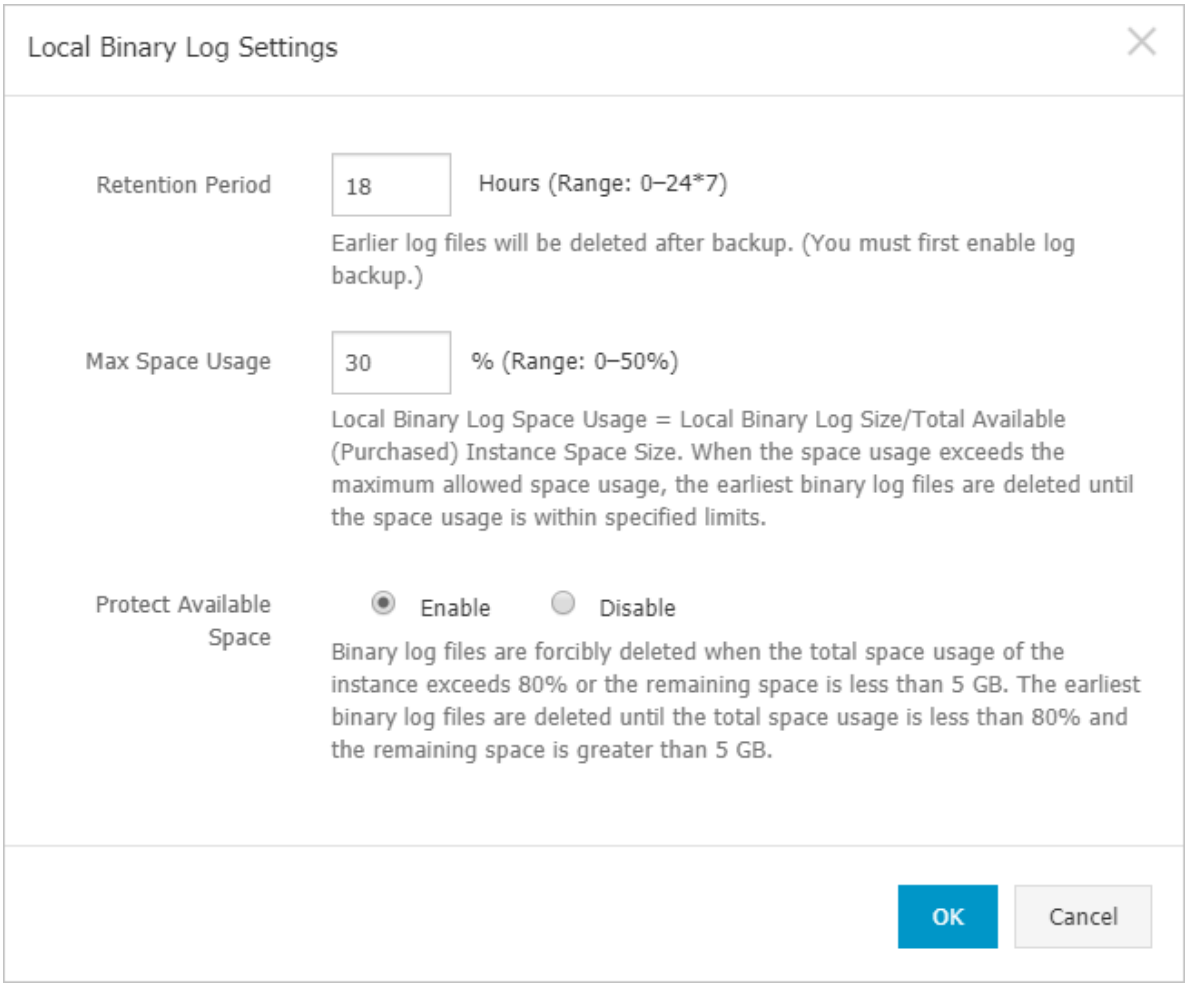
2. **In the upper-left corner, select the region where the target RDS instance is located.**



3. **In the left-side navigation pane, click Backup and Restoration.**
4. **Click the Local Log Settings tab and in the Local Binary Log Settings section view the local log settings of the RDS instance.**



5. **Click Edit.**



6. **In the Local Binary Log Settings dialog box, set Retention Period, Max Space Usage, and Protection Available Space**

   Parameter description:

   - **Retention Period: The period for retaining each generated binary log file. After the retention period elapses, the system deletes the binary log file after backing up it. Note that the system backs up binary log files only after you enable the log backup function. Value range: `0 - 7*24`. Default value: `18`. Unit: hour.**
   - **Max Space Usage: The maximum space usage that is allowed for local binary log files. When the space occupied by local binary log files exceeds the maximum usage, the system starts to delete earlier local binary log files. The**

system stops deleting local binary log files until the space usage drops below the maximum usage. Value range: `0 - 50%`. Default value: `30%`.

- Protection Available Space: Indicates whether the system deletes earlier local binary log files when the total space usage of the RDS instance exceeds 80% or the remaining space is less than 5 GB. The system stops deleting local binary log files when the total space usage drops below 80% and the remaining space is greater than 5 GB. Default value: `Enable`.

7. After you set the parameters, click OK.

16.5 Back up data across regions

This topic describes the cross-region backup function of RDS for MySQL. This function automatically replicates local backup files to an OSS bucket that is located in a specified destination region other than the region where your RDS instance is located. The backup files in the destination region are used to manage and restore your RDS instance.

Note:

- The cross-region backup function is different from the default backup function. For more information about the default backup function, see *Back up the data of an RDS for MySQL instance*.
- You can use a cross-region backup file to restore data to a new RDS instance in the destination region. For more information, see *Restore the data of an RDS for MySQL instance across regions*.

Differences between cross-region backup and default backup

- The cross-region backup function is disabled by default, and you must enable it manually. The default backup function is enabled by default.
- The cross-region backup function stores data to a different region from your RDS instance. The default backup function stores data the region where the instance is located.
- The cross-region backup function restores data to a new RDS instance in the source or destination region. The default backup function restores data to the source region where your RDS instance is located. For more information, see *Restore the data of an RDS for MySQL instance*.

- **The cross-region backup function is independent of your RDS instance. It retains the instance data based on the specified retention period even after you release the RDS instance. The default backup function retains the instance only for seven days after you release the RDS instance.**

Prerequisites

The RDS instance version is one of the following:

- **MySQL 5.7 High-availability Edition (with local SSDs)**
- **MySQL 5.6**

Billing

The fees for the cross-region backup function include the following two parts:

- **Fees for remote storage: 0.0002 USD/GB/hour.**
- **Traffic fees: For more information, see *Database Backup pricing*.**

Note:
In the Beta testing phase, you are charged only the traffic fees.

Precautions

- **The cross-region backup function restores data to a new RDS instance in the source or destination region.**
- **The cross-region backup function can only restore data to a new RDS instance but not to the original RDS instance.**
- **The cross-region backup function does not affect the default backup function. This means that cross-region backup files and default backup files can co-exist.**
- **With the cross-region backup function enabled, a backup is triggered when no valid backup set is generated within the last 24 hours.**
- **With the cross-region log backup function enabled, the system checks whether a valid backup set generated within the last 24 hours is available:**
  - **If continuous binary logs are generated following a valid backup set, the system dumps these binary logs.**
  - **If no continuous binary logs are generated following a valid backup set, the system starts a backup.**

- **The cross-region backup function is unavailable in some regions due to network resonances. The following table describes the destination regions available to each source region.**

Source region	Available destination regions
China (Hangzhou)	China (Shanghai), China (Qingdao), China (Shenzhen)
China (Shanghai)	China (Hangzhou), China (Qingdao), China (Shenzhen)
China (Qingdao)	China (Hangzhou), China (Shanghai), China (Shenzhen)
China (Beijing)	China (Hangzhou), China (Shanghai), China (Qingdao), China (Shenzhen)
China (Shenzhen)	China (Hangzhou), China (Shanghai), China (Qingdao)
China (Hong Kong)	China (Hangzhou), China (Shanghai), China (Qingdao), China (Shenzhen)

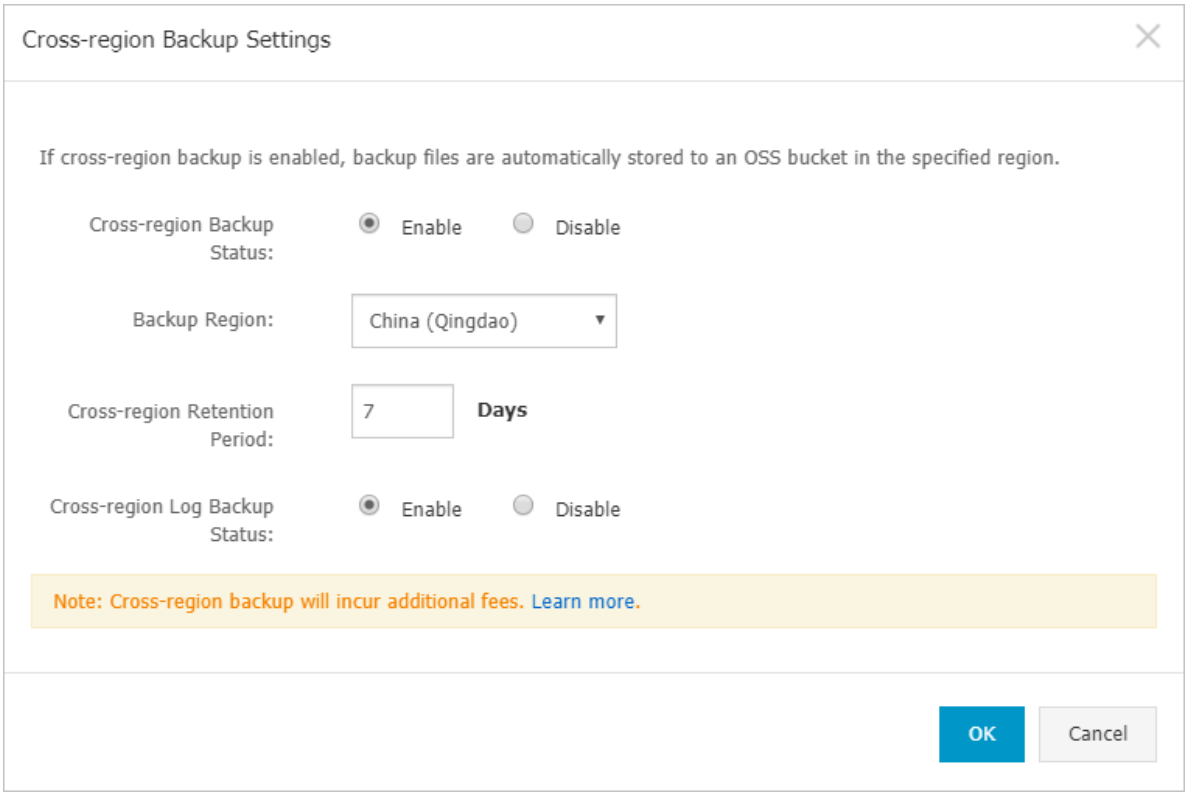
Method 1: Enable cross-region backup

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**

3. **Find the target RDS instance and in the Actions column choose More > Cross-region Backup Settings.**
4. **Set the following parameters.**

Parameter	Description
Cross-region Backup Status	The status of the cross-region backup function. Select Enable.

Parameter	Description
Backup Region	The region where backup files are stored. Local backup files are automatically replicated to an OSS bucket in the region.
Cross-region Retention Period	The period in which cross-region backup files are retained. The retention period spans from 7 days to 1,825 days, which is five years at most. **Note:** The system retains the cross-region backup files within the specified retention period even when the RDS instance expires or after you release the RDS instance. You can log on to the RDS console and go to the Cross-region Backup page to view the cross-region backup files that have not expired. For more information, see *Back up data across regions*.
Cross-region Log Backup Status	The status of the cross-region log backup function. If you enable this function, the system automatically replicates local log backup files to an OSS bucket in the specified destination region.



5. Click OK.

Method 2: Enable cross-region backup

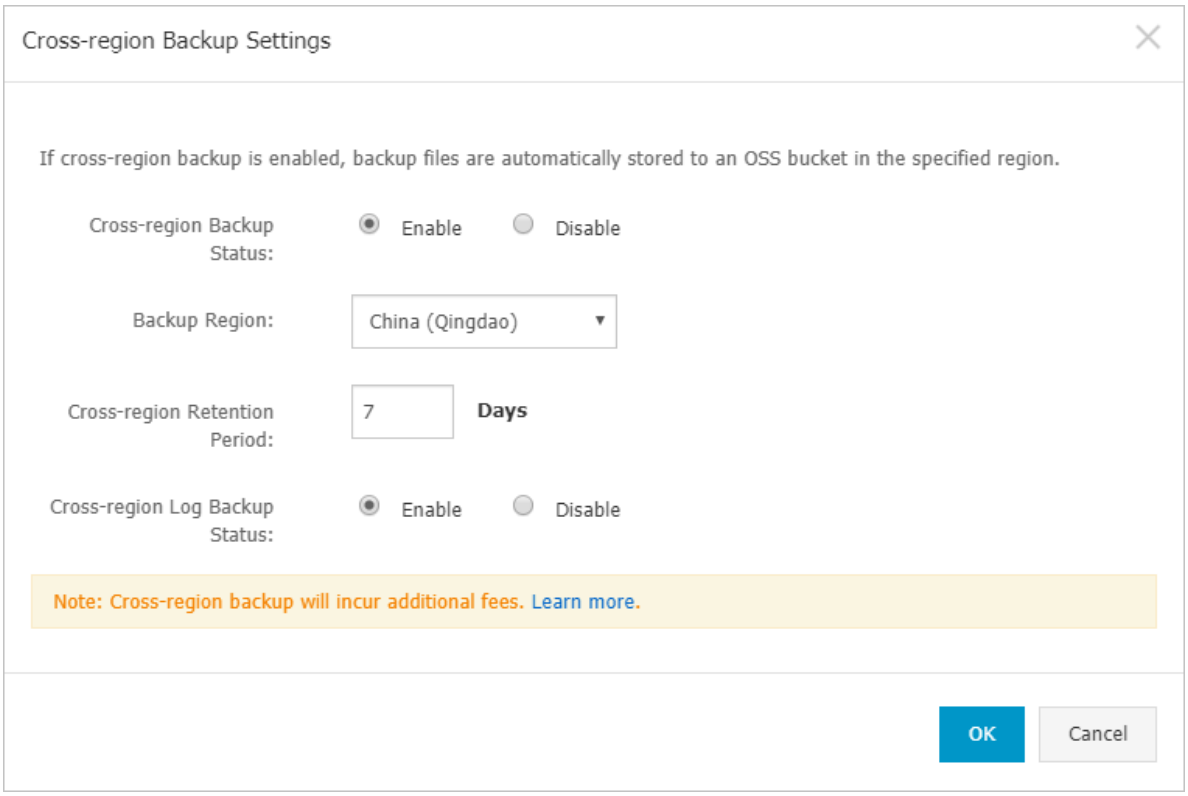
1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**
3. **Find the target RDS instance and click its ID.**
4. **In the left-side navigation pane, click Backup and Restoration.**
5. **On the Cross-region Backup tab, clickEdit.**

   Note:
   If the Cross-region Backup tab is not displayed, check whether all the conditions specified in the *Prerequisites* section are met.

6. **Set the following parameters:**

Parameter	Description
Cross-region Log Backup Status	The status of the cross-region backup function. Select Enable.
Backup Region	The region where backup files are stored. Local backup files are automatically replicated to an OSS bucket in the region.
Cross-region Retention Period	The period in which cross-region backup files are retained. The retention period spans from 7 days to 1,825 days, which is five years at most. **Note:** The system retains the cross-region backup files within the specified retention period even when the RDS instance expires or after you release the RDS instance. You can log on to the RDS console and go to the Cross-region Backup page to view the cross-region backup files that have not expired. For more information, see *Back up data across regions*.

Parameter	Description
Cross-region Log Backup Status	The status of the cross-region log backup function. If you enable this function, the system automatically replicates local log backup files to an OSS bucket in the specified destination region.



7. Click OK.

Modify cross-region backup settings

A menu item named Cross-region Backup is added to the left-side navigation pane on the RDS home page. With this menu item, you can modify cross-region backup settings even after your RDS instance is released.

1. Log on to the *RDS console*.
2. In the left-side navigation pane, click Cross-region Backup.
3. Find the target RDS instance and in the Cross-region Backup Settings column click Edit. Then modify the cross-region backup settings and click OK.

> **Note:**
> If the RDS instance has been released, then you can reconfigure only the Cross-region Retention Period parameter.

APIs

API	Description
#unique_191	Used to check whether an RDS instance has a cross-region backup set that can be used for cross-region restoration.
#unique_192	Used to restore data to a new RDS instance in the specified destination region.
#unique_193	Used to modify cross-region backup settings.
#unique_194	Used to query cross-region backup settings.
#unique_195	Used to list cross-region data backup files.
#unique_196	Used to list cross-region log backup files .
#unique_197	Used to query the destination regions available to a specified region.
#unique_198	Used to query the time period to which a selected backup file can restore an RDS instance.
#unique_199	Used to query the cross-region backup settings in a specified region.

17 Database restoration

17.1 Restore the data of an RDS for MySQL instance

This topic describes how to restore the data of an RDS for MySQL instance by using a data backup.

You can use one of the following methods to restore the data of an RDS for MySQL instance:

- **Method 1: Restore data to a new RDS instance, verify the data, and then migrate the data to the original RDS instance. This topic introduces this method.**
- **Method 2: Restore single-database or -table data to the original RDS instance or a new RDS instance. For more information, see** *Restore individual databases or tables for an RDS for MySQL instance***.**

Precautions

- **The whitelist settings, backup settings, and parameter settings of the new RDS instance must be the same as those of the original RDS instance.**
- **The data information of the new RDS instance must be the same as that of the used backup file or that from the specified time point.**
- **The new RDS instance carries the account information in the used backup file or that from the specified time point.**

Fees

For more information, see *ApsaraDB RDS for MySQL pricing***.**

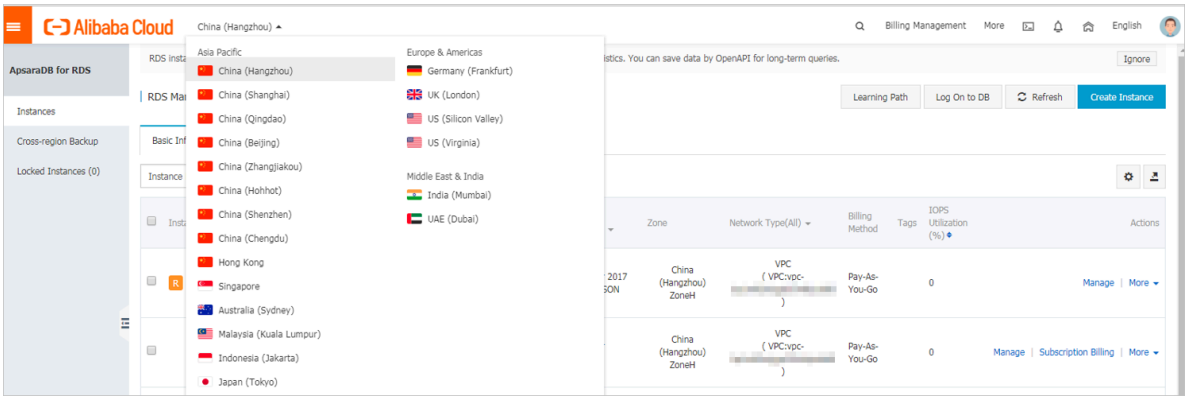
Prerequisites

The original RDS instance must meet the following conditions:

- **The instance is in the Running state and is not locked.**
- **No migration task is being performed for the instance.**
- **If you want to restore the data from a time point, the log backup function is enabled.**
- **If you want to restore the data from a backup set, at least one backup set is available for the instance.**

Restore data to a new RDS instance.

1. **Log on to the *RDS console*.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**



3. **Find the target RDS instance and click its ID.**
4. **In the left-side navigation pane, click Backup and Restoration.**
5. **In the upper-right corner, click Restore Database (Previously Clone Database).**
6. **On the displayed Restore Database (Previously Clone Instance) page, select a billing method:**
   - **Pay-As-You-Go: Fees are calculated by hour according to the actual job size. This billing method is suitable to a short-term RDS instance, which can be released immediately after you finish the data restoration.**
   - **Subscription: Fees are estimated in advance, and the relevant usage allocation is paid for when you create an RDS instance. This billing method is suitable to a long-term RDS instance, which is cheaper than a pay-as-you-go instance. Additionally, a longer duration of purchase indicates a higher discount rate.**

> **Note:**
> **You can change the billing method of an RDS instance from pay-as-you-go to subscription but not from subscription to pay-as-you-go.**

7. Set the parameters of the new RDS instance.

Parameter	Description
Restore Mode	· By Time: You can select any time point within the specified log retention period. For more information about how to view or change the log retention period, see *Back up the data of an RDS for MySQL instance*. · By Backup Set **Note:** The By Time option is available only when the log backup function is enabled.
Zone	A zone is a physical area within a region. Different zones in the same region are basically the same. You can create an RDS instance in the same or different zone from the corresponding ECS instance. **Note:** The new RDS instance must be located in the same region as the original RDS instance.
Edition	· Basic: The DB system has only one instance. In this edition, compute is separated from storage, which is cost-effective. However, we recommend that you do not use this edition in production environments. · High-availability: The DB system has two instances: one master instance and one slave instance. The two instances work in a classic high-availability architecture. · Edition: The DB system has three instances: one master instance and two slave instances. The three instances are located in three different zones in the same region to guarantee service availability. This edition is available to the China (Hangzhou), China (Shanghai), China (Shenzhen), and China (Beijing) regions. For more information about the product series, see *#unique_22*. The product series available in the console vary depending on the used DB engine version.

Parameter	Description
CPU and Memory	The type (including the CPU and memory specifications) of the new RDS instance. The CPU, memory, and storage capacity specifications of the new RDS instance must be higher than hose of the original RDS instance. Otherwise, the data restoration may take a long time. Each instance type supports a specific number of CPU cores, memory size, maximum number of connections, and maximum IOPS. For more information, see *#unique_24*. RDS instances fall into the following three type families: · General-purpose instance: A general-purpose instance owns dedicated memory and I/O resources, but shares CPU and storage resources with the other general-purpose instances on the same server. · Dedicated instance: A dedicated instance owns dedicated CPU, memory, storage, and I/O resources. · Dedicated host: A dedicated-host instance owns all the CPU, memory, storage, and I/O resources on the server where it is located. For example, 8 Cores, 32 GB indicates a general-purpose instance, 8 Cores, 32 GB (Dedicated Instance) indicates a dedicated instance, and 30 Cores, 220GB (Dedicated Host)30 Cores, 220 GB (Dedicated Host) indicates a dedicated-host instance.
Capacity	Used for storing data, system files, binlog files, and transaction files.
Network Type	· Classic Network: a classic network. · VPC (recommended): A VPC is an isolated network environment that provides better security and performance than a classic network.

8. Optional. If the new RDS instance uses the subscription billing method, set the Duration and Quantity parameters.
9. Click Buy Now.
10. On the Order Confirmation page, select Terms of Service, Service Level Agreement, and Terms of Use, then click Pay Now to complete the payment.

Verify data in the new RDS instance

For more information, see *Connect to an RDS for MySQL instance*.

Migrate data to the original RDS instance

After verifying the data in the new RDS instance, you can migrate the data to the original RDS instance.

Data migration refers to migrating data from one RDS instance (the source RDS instance) to another (the destination RDS instance). The data migration operation does not interrupt the source RDS instance.

Precautions

Do not perform DDL operations during the data migration. Otherwise, the data migration may fail.

Procedure

1. **Log on to the** *DTS console*.
2. **In the left-side navigation pane, click Data Migration.**
3. **In the upper-right corner, click Create Migration Task.**

4. **Enter the migration task name, source database information, and destination database information.**

   Parameter description:

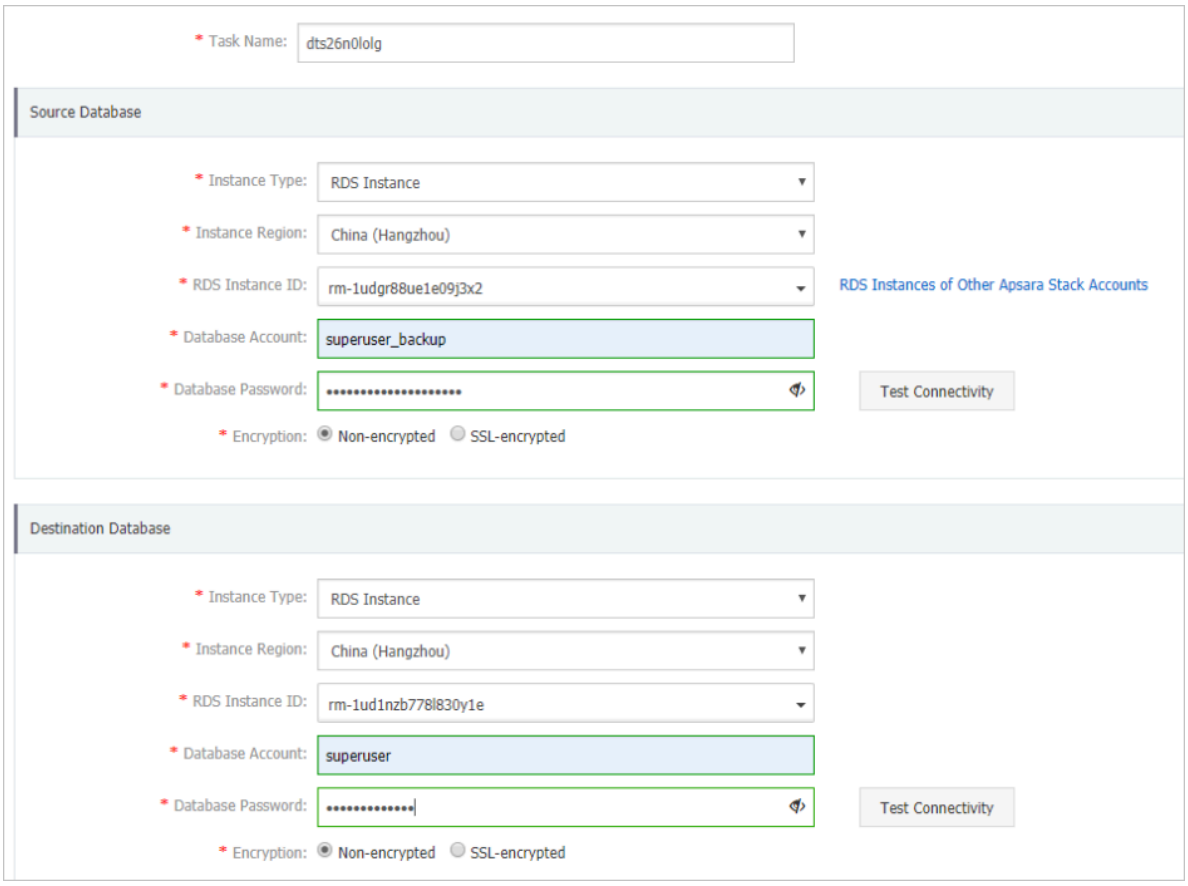
   - **Task Name: By default, DTS automatically generates a name for each migration task. You can change the name as needed.**
   - **Source Database**
     - **Instance Type: Select RDS Instance.**
     - **Instance Region: Select the region where the new RDS instance is located.**
     - **RDS Instance ID: Select the ID of the new RDS instance.**
     - **Database Account: Enter the username of the account for the new RDS instance.**
     - **Database Password: Enter the password of the account for the new RDS instance.**
     - **Connection: Select Non-encrypted. If the new RDS instance supports *SSL encryption* and has SSL encryption enabled, then you must select SSL-encrypted.**

     > **Note:**
     > **The values of the Instance Type and RDS Instance ID parameters determine which of the other parameters are displayed.**

   - **Destination Database**
     - **Instance Type: Select RDS Instance.**
     - **Instance Region: Select the region where the original RDS instance is located.**
     - **RDS Instance ID: Select the ID of the original RDS instance.**
     - **Database Account: Enter the username of the account for the original RDS instance.**
     - **Database Password: Enter the password of the account for the original RDS instance.**
     - **Connection: Select Non-encrypted. If the original RDS supports *SSL encryption* and has SSL encryption enabled, then you must selectSSL-encrypted.**

     > **Note:**

The values of the Instance Type and RDS Instance ID parameters determine which of the other parameters are displayed.



5. **Click Set Whitelist and Next.**
6. **Select Schema Migration and Full Data Migration next to Migration Types.**
7. **In the Available section, select the objects you want to migrate. Then click > to move the selected objects to the Selected section.**

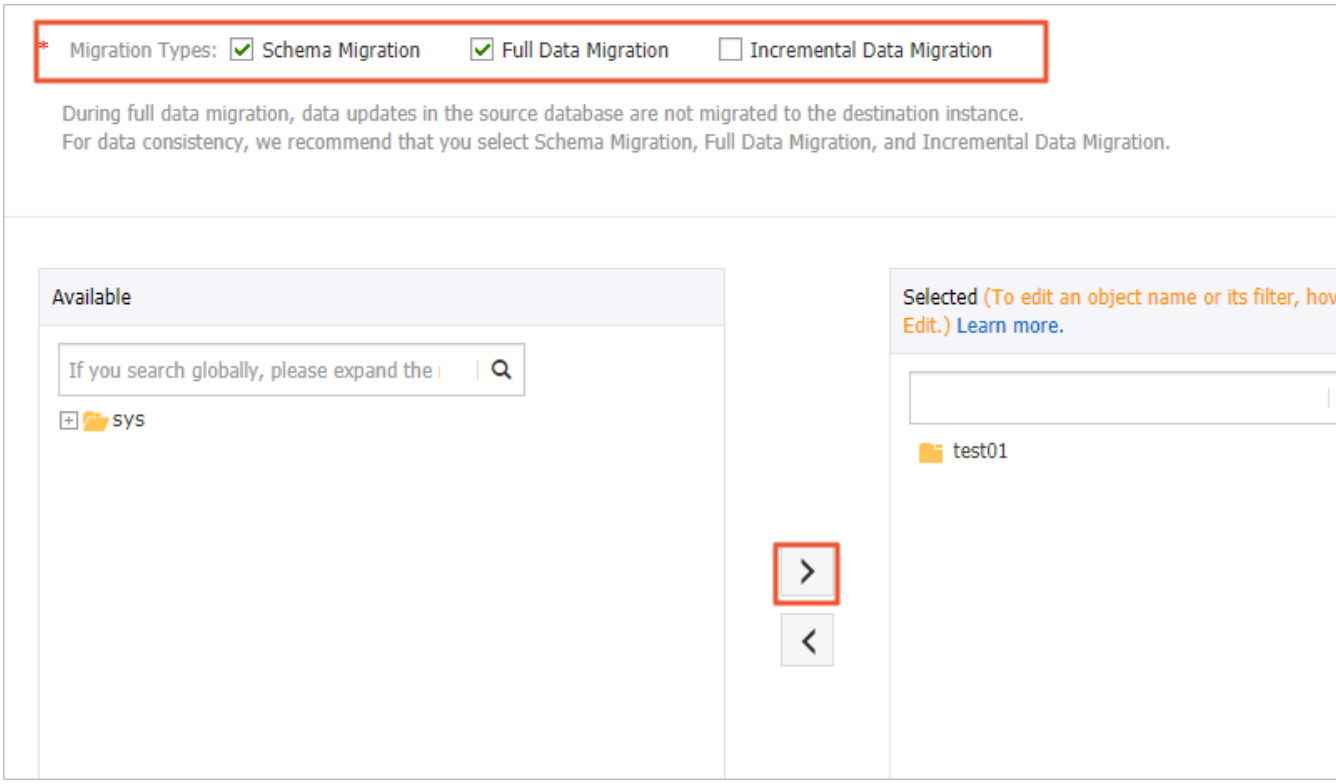
Note:

DTS checks for objects that have the same name. If the destination RDS instance has an object whose name is the same as the name of an object to be migrated, the data migration fails.

In such case, take one of the following two operations:

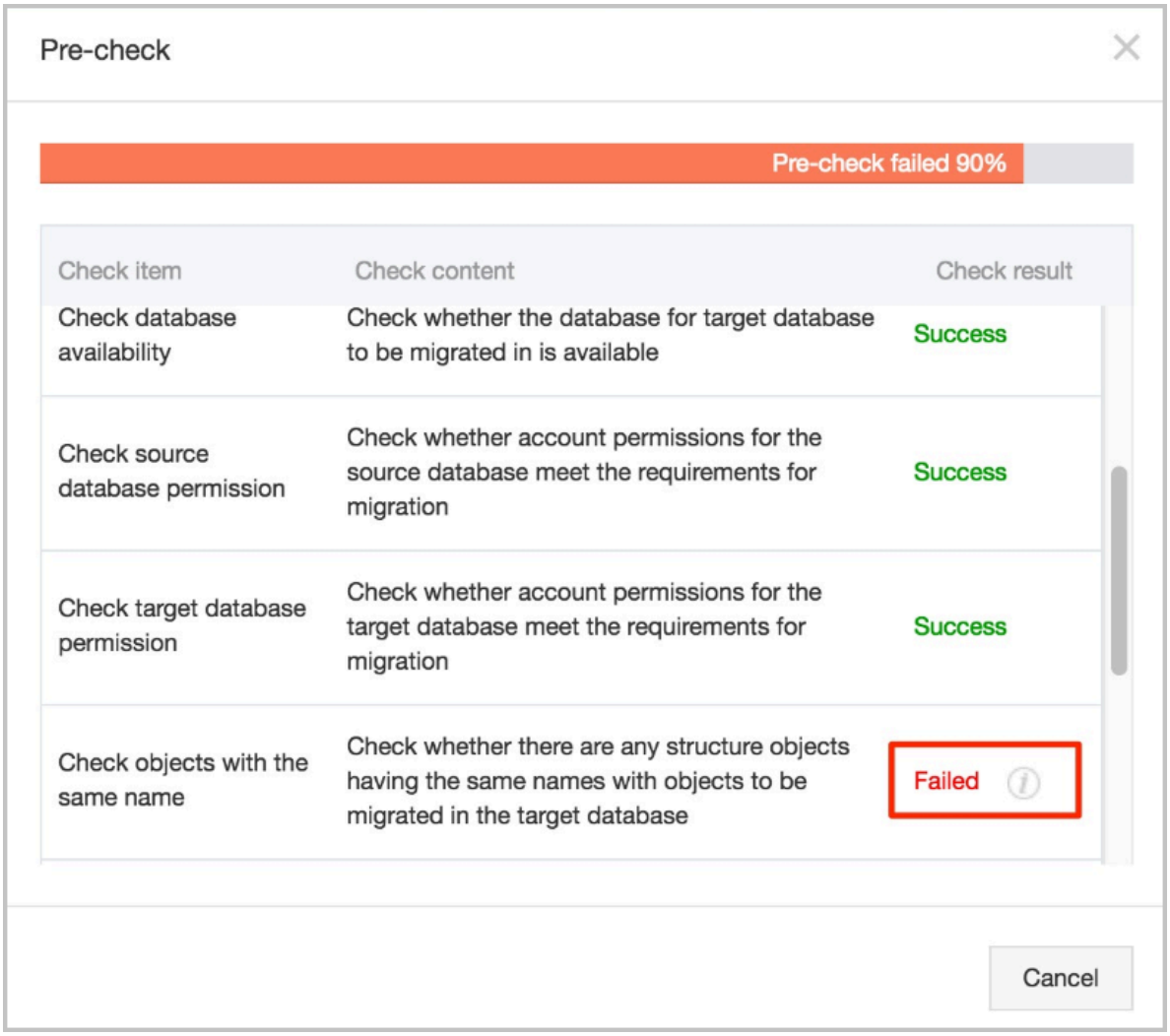
- **In the Selected section, move the pointer over the object whose name you want to change, click Edit, and in the displayed dialog box enter the new object name.**

- **Rename the object in the destination RDS instance.**

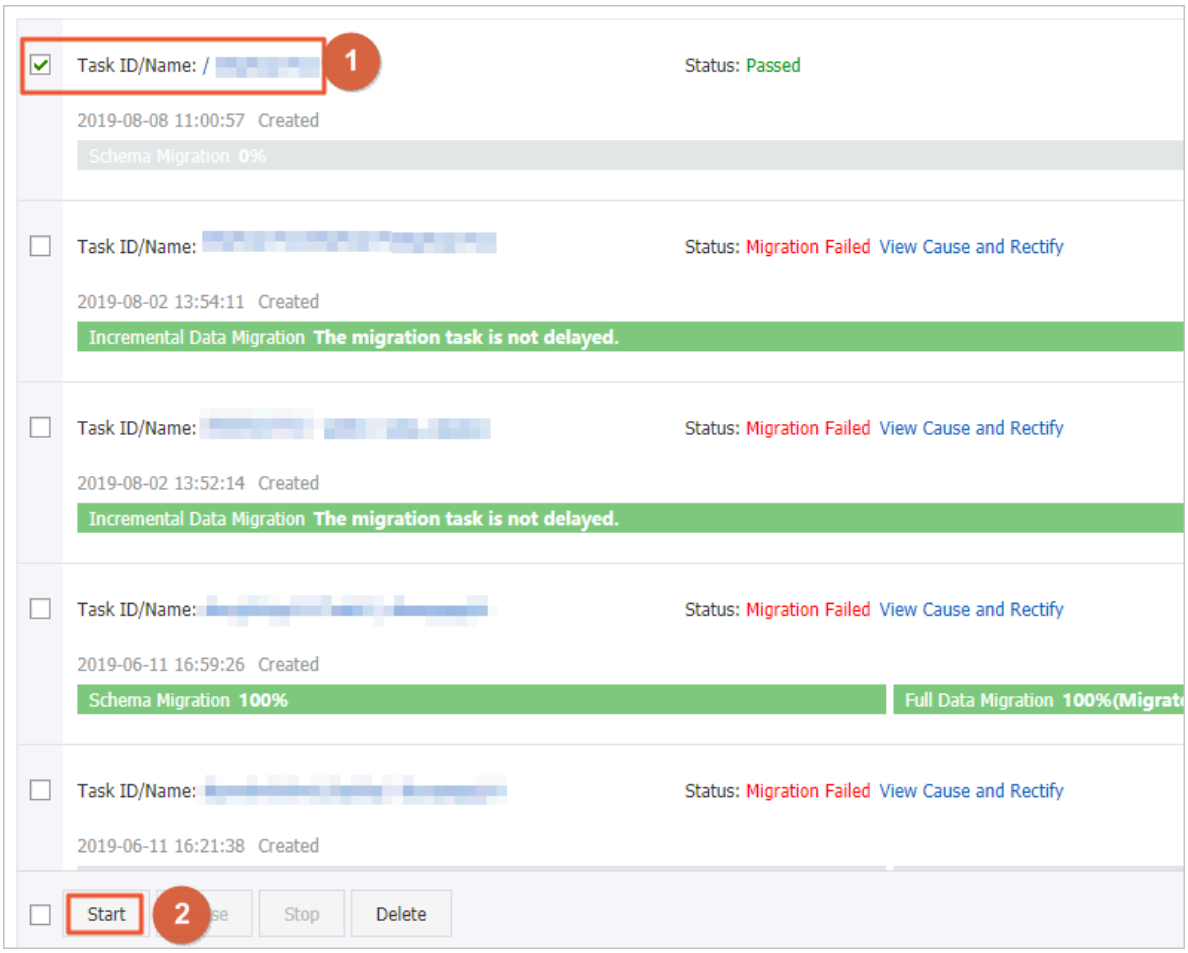


8. **Click Precheck.**

9. **Optional. If the migration task fails the precheck, click ⓘ next to the check item whose Result is Failed, and resolve the problem according to the failure information.**



10. On the page that displays migration tasks, select the migration task you created, then click Start.



11. When the migration task passes the precheck, click Next.
12. In the Confirm Settings dialog box, confirm the configuration, select Data Transmission Service (Pay-As-You-Go) Service Terms, and click Buy and Start.

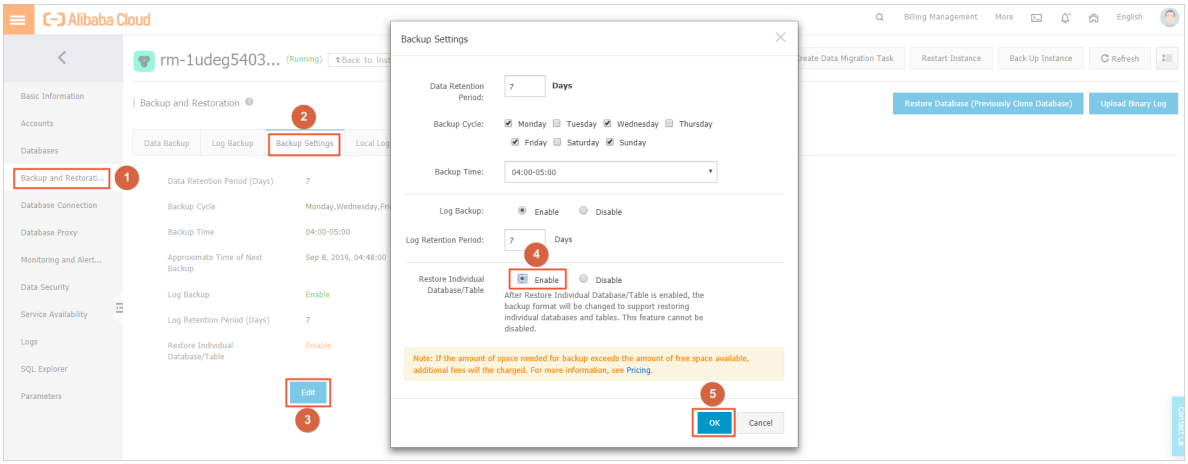
17.2 Restore individual databases or tables for an RDS for MySQL instance

This topic describes how to restore individual databases or tables for an RDS for MySQL instance.

Prerequisites

- **The DB engine version and edition is one of the following:**
  - **MySQL 5.6 High-availability Edition**
  - **MySQL 5.7 High-availability Edition (with local SSDs)**

- The Restore Individual Database/Table function is enabled in the console (navigation path: Backup and Restoration > Backup Settings).



Note:

After the Restore Individual Database/Table function is enabled, the backup file format changes to support database and table restoration. This function cannot be disabled once it is enabled.

- If you want to restore data to the current RDS instance, the current RDS instance must meet the following conditions:
  - The instance is in Running state and is not locked.
  - The instance is not involved in any migration task.
  - If you want to restore data by time, the log backup function is enabled.
  - If you want to restore data by backup set, the instance must have at least one backup set.
- If you want to restore data to a new RDS instance, the current RDS instance must meet the following conditions:
  - The instance is in Running state and is not locked.
  - If you want to restore data by time, the log backup function is enabled.
  - If you want to restore data by backup set, the instance must have at least one backup set.

Precautions

- The Restore Individual Database/Table function changes the format of backup files from .tar to .xbstream. After this change, the storage space occupied by backup files slightly increases. If the occupied storage space exceeds the quota of

free backup space, additional fees are incurred. For more information, see *View the quota of free backup space for an RDS for MySQL instance*. **We recommend that you plan the backup cycle properly to meet your business needs while also maximizing backup space utilization.**

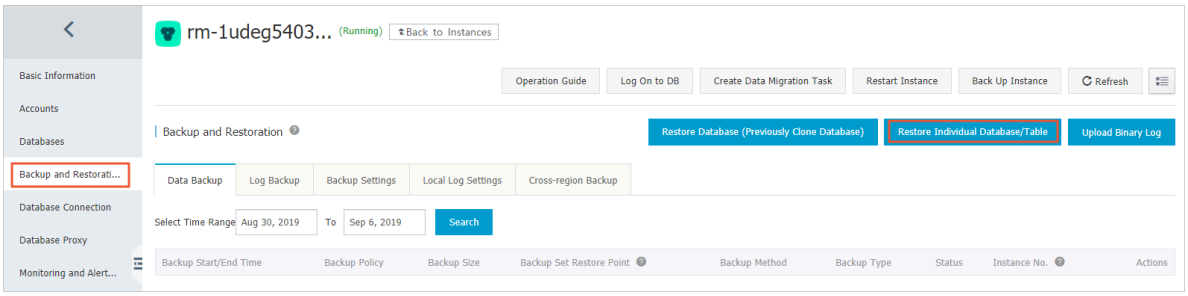
- **The Restore Individual Database/Table function is available only when the number of tables in the RDS instance does not exceed 50,000.**
- **You can select and restore up to 50 databases or tables.**

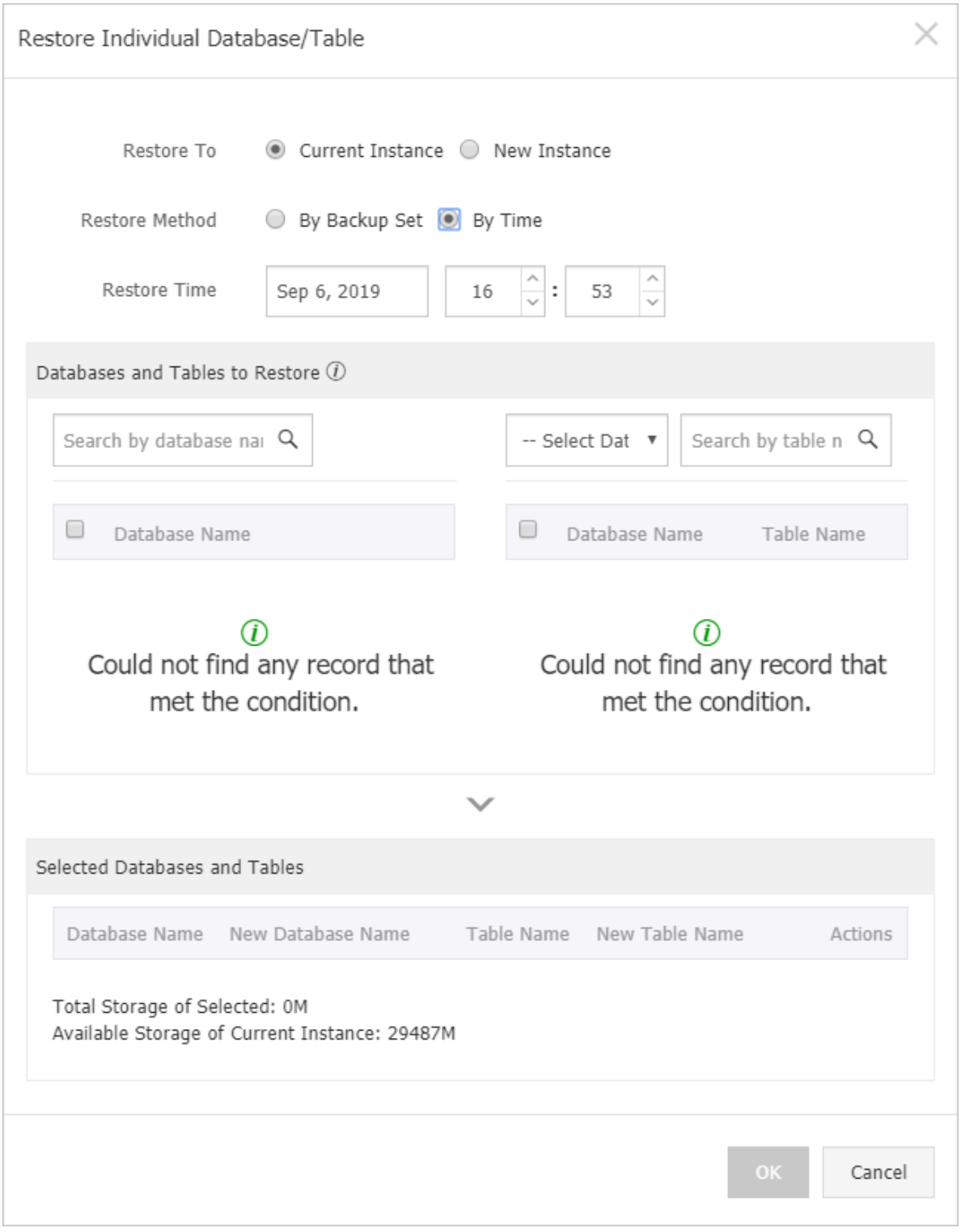
Procedure

1. **Log on to the** *RDS console*.
2. **In the upper-left corner, select the region where the target RDS instance is located.**
3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Backup and Restoration.**
5. **In the upper-right corner, click Restore Individual Database/Table. In the displayed dialog box, set the following parameters.**

Note:

If the Restore Individual Database/Table button is unavailable, you must enable the Restore Individual Database/Table function. For more information, see *Prerequisites*.





Parameter	Description
Restore To	· Current Instance: restores the selected databases or tables to the current RDS instance. · New Instance: purchases a new RDS instance and restores the selected databases or tables to the new RDS instance.

Parameter	Description
Restore Method	· By Backup Set · By Time: You can specify any time point within the log retention period. For information about how to view or change the log retention period, see *Back up the data of an RDS for MySQL instance*. **Note:** The By Time option is available only when the log backup function is enabled.
Backup Set	Select the backup set used for restoring databases or tables. **Note:** This parameter is available when you set the Restore Method parameter to By Backup Set.
Restore Time	Select the time point from which you want to restore databases or tables. **Note:** This parameter is available when you set the Restore Method parameter to By Time.
Databases and Tables to Restore	Select the databases or tables you want to restore.
Selected Databases and Tables	· Displays the selected databases or tables. You can specify the names of the databases or tables to which data is restored. · Displays the total size of the selected databases or tables, and the remaining storage space of the instance. Make sure that the remaining storage space is sufficient.

6. Click OK.

Note:

If you set Restore To to New Instance, you are directed to the page for purchasing a new RDS instance. You must set the instance parameters and complete the payment.

Parameter	Description
Edition	· Basic: The DB system has only one instance. In this edition, compute is separated from storage, which is cost-effective. However, we recommend that you do not use this edition in production environments. · High-availability: The DB system has two instances: one master instance and one slave instance. The two instances work in a classic high-availability architecture. · Edition: The DB system has three instances: one master instance and two slave instances. The three instances are located in three different zones in the same region to guarantee service availability. This edition is available to the China (Hangzhou), China (Shanghai), China (Shenzhen), and China (Beijing) regions. For more information about the product series, see *#unique_22*. The product series available in the console vary depending on the used DB engine version.
Zone	A zone is an independent physical location in a region. Zones in the same region are basically the same. You can create an RDS instance in the same or different zone from your ECS instance. Note: The new RDS instance must be in the same region as the current RDS instance.

Parameter	Description
CPU and Memory	Each instance type corresponds to specific number of CPU cores, memory space, maximum number of connections, and maximum IOPS. For more information, see *#unique_24*. RDS instances fall into the following types: · General-purpose instance: A general-purpose instance owns dedicated memory and I/O resources but shares CPU and storage resources with the other general-purpose instances on the same server. · Dedicated instance: A dedicated instance owns dedicated CPU, memory, storage, and I/O resources. · Dedicated host: A dedicated-host instance owns all the CPU, memory, storage, and I/O resources on the server where it is deployed. For example, 8 Cores, 32 GB is the specification for a general-purpose instance, 8 Cores, 32 GB (Dedicated instance) is the specification for a dedicated instance, and 30 Cores, 220 GB (Dedicated host) is the specification for a dedicated-host instance.
Capacity	Used to store data files, system files, binary log files, and transaction files.
Network Type	· Classic Network: a classic network. · VPC (recommend): A Virtual Private Cloud (VPC) is an isolated network that is superior to a classic network in terms of security and performance.

17.3 Restore the data of an RDS for MySQL instance across regions

This topic describes how to restore the data of an RDS for MySQL instance across regions. Specifically, you can restore data to a new RDS instance in the source or destination region.

Prerequisites

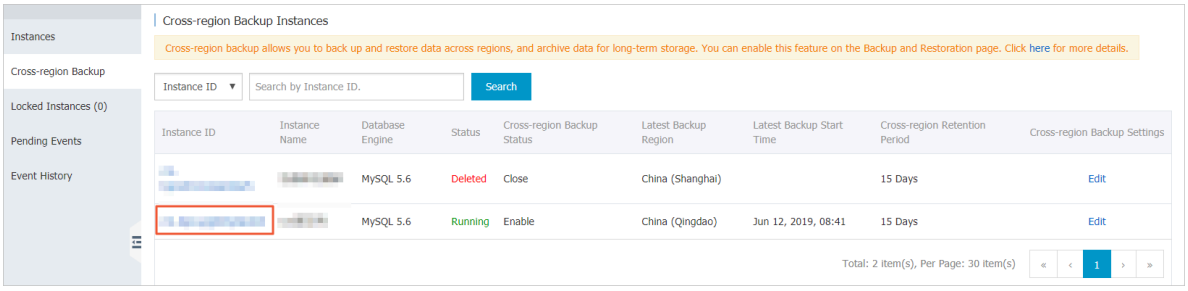
You have completed a cross-region backup. For more information, see *Back up data across regions*.

Precautions

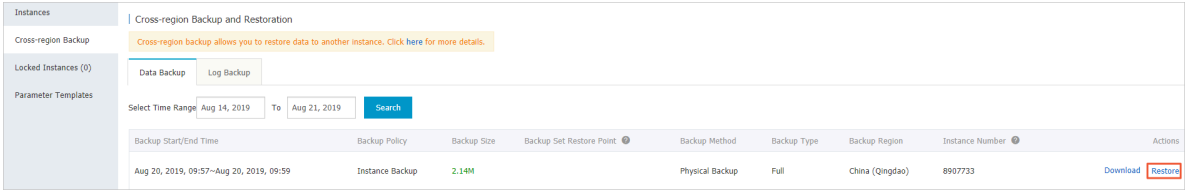
- If the original RDS instance has the database proxy enabled and does not have a superuser account, you may need to reset the password to connect to the new RDS instance to which data is restored.
- You cannot restore data to the original RDS instance by using a cross-region backup file.

Procedure

1. Log on to the *RDS console*.
2. In the left-side navigation pane, click Cross-region Backup.
3. Find the target RDS instance and click its ID.



4. On the Data Backup tab, find the target backup set and in the Actions column click Restore.



5. On the Restore Database page, click the Subscription or Pay-As-You-Go tab and set the following parameters.

Parameter	Description
Restore Mode	• By Backup Set: You can restore the data in the selected backup set to the new RDS instance. • By Time: You can select any time point within the log backup retention period to restore the data from this time point to the new RDS instance.
Backup Set	When the Restore Mode parameter is set to By Backup Set, you must select a backup set for data restoration.

Parameter	Description
Restore Point	When the Restore Mode parameter is set to By Time, you must select a time point from which you want to restore data. **Note:** Both local and cross-region log backup sets can be used for time-based data restoration.
Region	The ID of the region where the new RDS instance is located. You can use a cross-region backup set to restore data only to a new RDS instance in the source or destination region.
Zone	An independent physical area within a region. Different zones in the same region are basically the same. The new RDS instance can be located in the same region or in a different region from the ECS instance.
CPU and Memory	The type (including CPU and memory specifications) of the new RDS instance. Each instance type supports a specific number of CPU cores, memory size, maximum number of connections, and maximum IOPS. For more information, see *#unique_24*.
Capacity	Used for storing data, system files, binary log files, and transaction files.
Network Type	· Classic Network: a classic network. · VPC (recommended): A VPC is an isolated network environment that provides better security and performance than a classic network. If you select VPC, you must also select a VSwitch.

6. Optional. If the new RDS instance uses the Subscription billing method, set the Duration and Quantity parameters.
7. Click Buy Now.
8. On the Order Confirmation page, select Terms of Service, Service Level Agreement, and Terms of Use, then click Pay Now to complete the payment.

What to do next

In the upper-left corner of the RDS console, select the region where the new RDS instance is located, then you can view the new RDS instance.

After the new RDS instance is created, you must *configure a whitelist* **and** *create accounts*. **If you want to connect to the new RDS instance through the Internet, you must also** *apply for a public IP address*. **Then you can** *connect to the new RDS instance*.

APIs

API	Description
#unique_191	Used to check whether an RDS instance has a cross-region backup set that can be used for cross-region restoration.
#unique_192	Used to restore data to a new RDS instance in the specified destination region.
#unique_193	Used to modify cross-region backup settings.
#unique_194	Used to query cross-region backup settings.
#unique_195	Used to list cross-region data backup files.
#unique_196	Used to list cross-region log backup files .
#unique_197	Used to query the destination regions available to a specified region.
#unique_198	Used to query the time period to which a selected backup file can restore an RDS instance.
#unique_199	Used to query the cross-region backup settings in a specified region.

18 Disable the database proxy mode

This topic describes how to disable the database proxy mode for an RDS for MySQL instance. Disabling the database proxy mode means switching to the standard mode, which helps improve the performance of the RDS instance.

Notice:
The database proxy mode may cause service instability in certain circumstances. For smooth service operation, we recommend that you upgrade the network connection mode of your RDS instance as soon as possible. For more information, see *#unique_203*.

Precautions

- **In the database proxy mode, the multi-statement function is enabled by default at the protocol layer. Therefore, after you disable the database proxy mode, if you do not enable the multi-statement function but run multiple SQL statements, the system reports errors in the SQL statements. To prevent this problem, you must check and add connection parameters in advance. For example, you can add the `allowMultiQueries` parameter to JDBC as follows:**

```
dbc:mysql:///test?allowMultiQueries=true
```

- **You can only disable the database proxy mode (that is, switch from the database proxy mode to the standard mode). You cannot enable the database proxy mode (that is, switch from the standard mode to the database proxy mode).**
- **Switching the access mode may cause a 30-second transient disconnection. Therefore, we recommend that you switch the access mode during off-peak hours or make sure that your application can automatically reconnect to the RDS instance.**

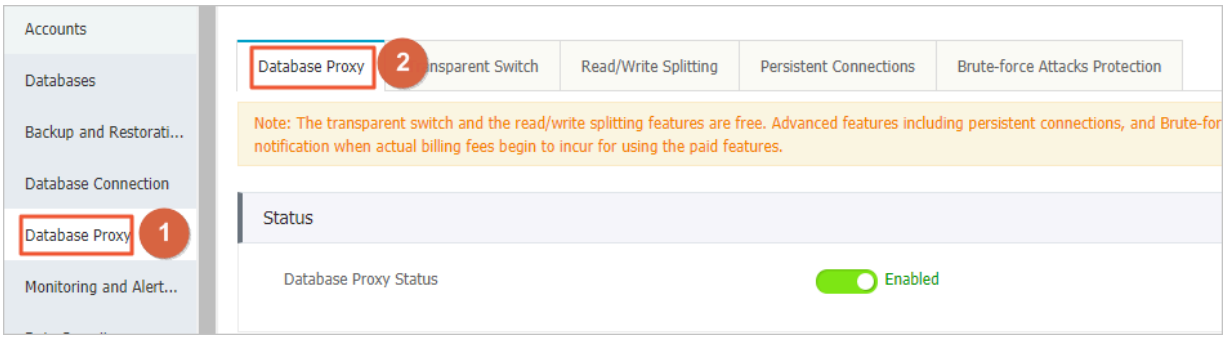
Prerequisites

The database proxy mode is enabled for your RDS instance.

Note:

- **If the Database Proxy tab is displayed, the database proxy mode is enabled and you can proceed with the operations described in this topic.**

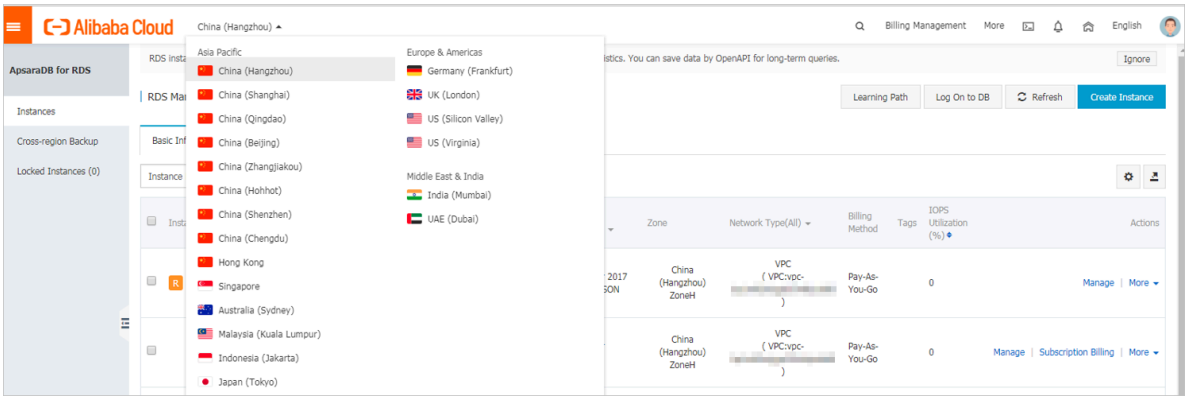
- If the Database Proxy tab is not displayed, the database proxy mode is not displayed and you can skip this topic.



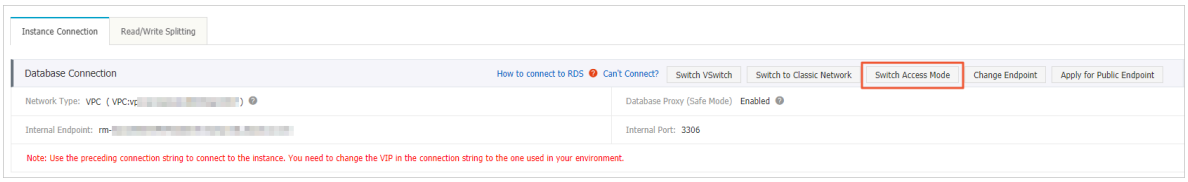
Procedure

Method 1

1. Log on to the *RDS console*.
2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Database Connection.
5. Click Switch Access Mode and in the displayed dialog box, click Confirm.



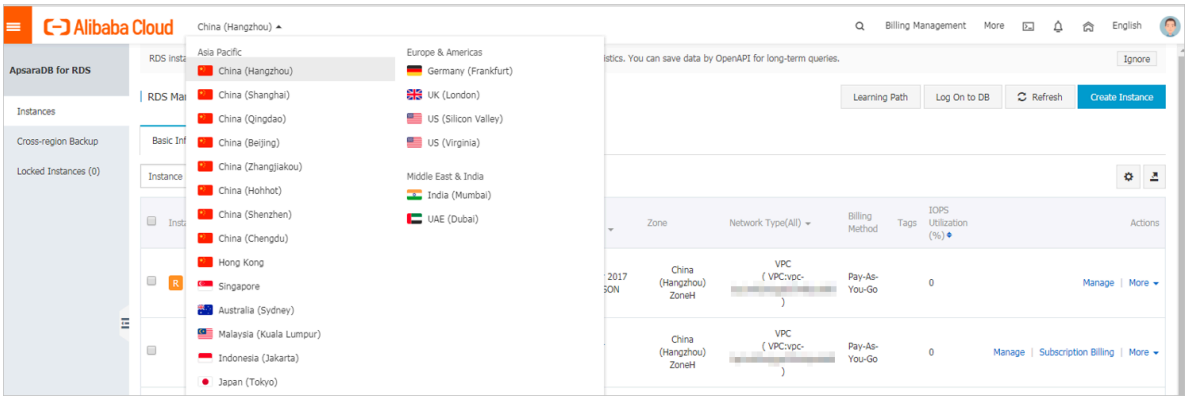
Note:

This button is available only when you have enabled the database proxy mode.

Method 2

1. Log on to the *RDS console*.

2. In the upper-left corner, select the region where the target RDS instance is located.



3. Find the target RDS instance and click the instance ID.
4. In the left-side navigation pane, click Database Proxy.
5. On the Database Proxy tab, click the slider next to the database proxy status and in the displayed dialog box, click Confirm.

> **Note:**
> This tab page is available only when you have enabled the database proxy mode.

19 Manage logs

This topic describes how to manage logs through the RDS console or by using SQL statements. You can query error logs and slow query logs. The log query results help you to locate faults. All RDS for MySQL instances except those in the Basic Edition support log management.

- **For information about log backup policies and rules, see** *Back up the data of an RDS for MySQL instance*.
- **For information about how to download log backup files, see** *Download the data backup files and log backup files of an RDS for MySQL instance*.
- **For information about how to restore data through log backup files, see:**
  - *Restore the data of an RDS for MySQL instance*
  - *#unique_187*

Procedure

1. **Log on to the** *RDS console*.
2. **In the upper-left corner, select the region where the target RDS instance is located.**

3. **Find the target RDS instance and click the instance ID.**
4. **In the left-side navigation pane, click Log Management.**

5. On the Log Management page, select Error Log, Slow Query Log, Slow Query Log Summary, or Primary/Secondary Instance Switch Log, select a time range, and click Search.

Query item	Description
Error Log	Records the SQL statements that are failed to be executed within the last one month.
Slow Query Log	Records the SQL statements that lasted for more than 1 second within the last one month. (You can reconfigure the `long_query_time` parameter to change this time threshold according to *Reconfigure parameters for an RDS for MySQL instance*.) Similar SQL statements are displayed once only. **Note:** The slow query logs are updated in the console once every minute. You can view the `mysql.slow_log` table to obtain real-time slow query logs.
Slow Query Log Summary	Provides statistics and analysis reports for SQL statements that lasted for more than 1 second within the last one month. (You can reconfigure the `long_query_time` parameter to change this time threshold according to *Reconfigure parameters for an RDS for MySQL instance*.)
Primary/Secondary Instance Switch Log	Available to RDS instances in the MySQL High-availability Edition.

Note:
For each RDS instance in the China (Zhangjiakou) region, the system retains only the error logs, slow query logs, and slow query log summary generated within the last nine days.

20 Tag management

20.1 Create tags

This topic describes how to create tags for one or more RDS instances. If you have a large number of RDS instances, you can create tags and then bind the tags to the instances so that you can classify and better manage the instances. Each tag consists of a key and a value.

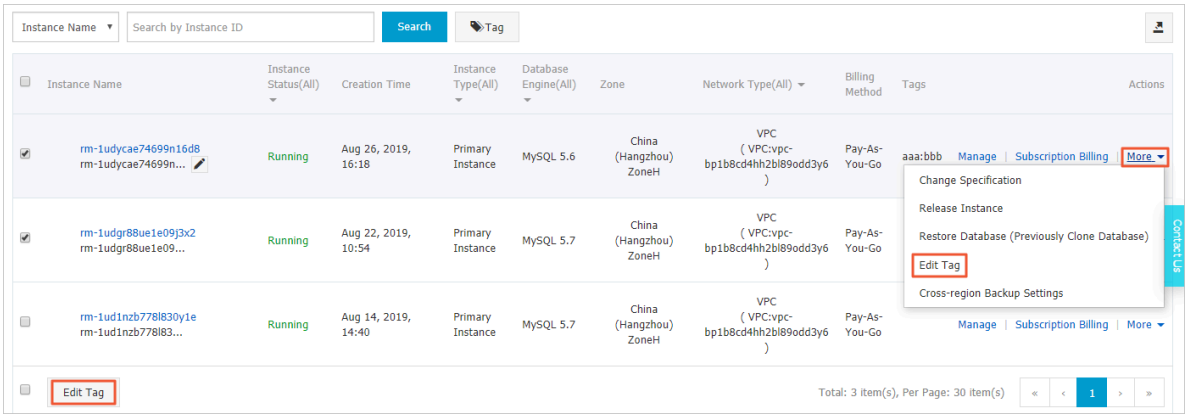
Limits

- **Up to 10 tags can be bound to each RDS instance, and each tag must have a unique key. Tags with the same key are overwritten.**
- **You can bind up to five tags at a time.**
- **Tag information is independent in different regions.**
- **After you unbind a tag from an RDS instance, the tag is deleted if it is not bound to any other RDS instance.**

Procedure

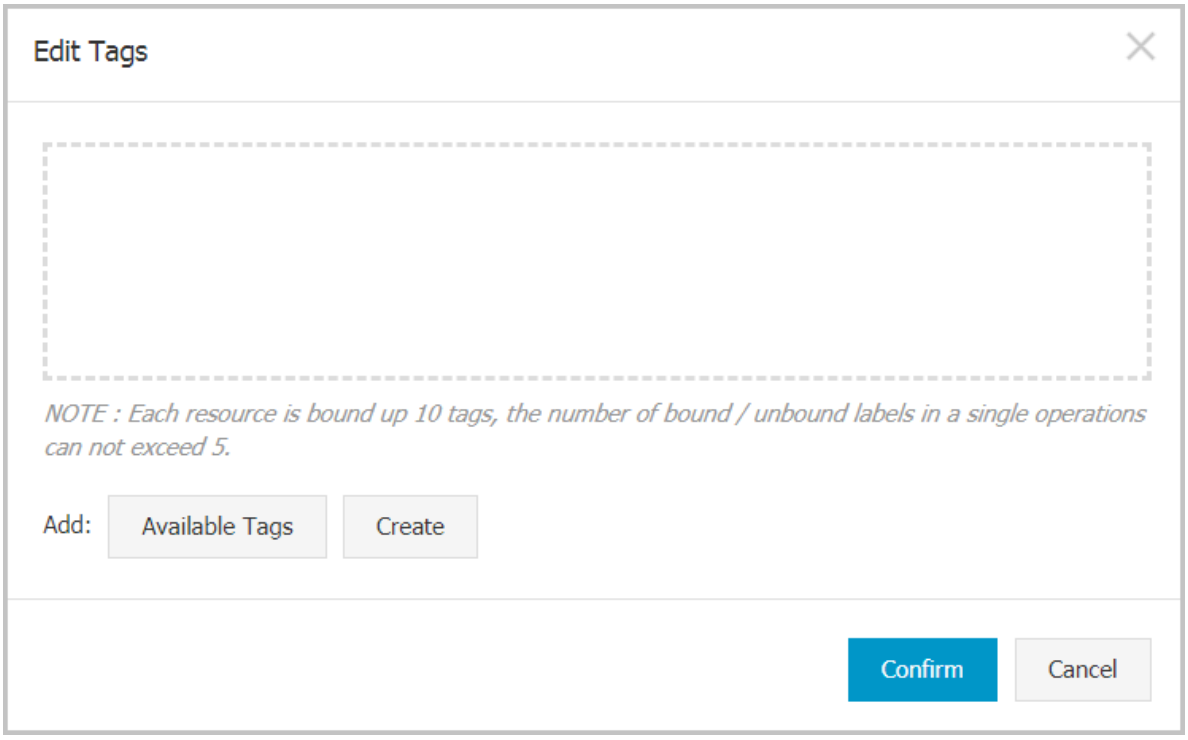
1. **Log on to the *RDS console* and in the left-side navigation pane, click Instances.**
2. **In the upper-left corner, select the region where the target RDS instance is located.**

3. Specify the method of adding tags.
   - If you want to add tags to only one RDS instance, find the RDS instance and in the Actions column choose More > Edit Tag.
   - If you want to add tags to more than one RDS instance, select the RDS instances and click Edit Tag



4. Click Add, enter the Key and Value, and click Confirm.

> **Note:**
> If you have already created tags, you can click Available Tags and select an existing tag.



5. After you add all the tags you need, click Confirm.

APIs

API	Description
AddTagsToResource	Used to bind a tag to RDS instances.

20.2 Delete tags

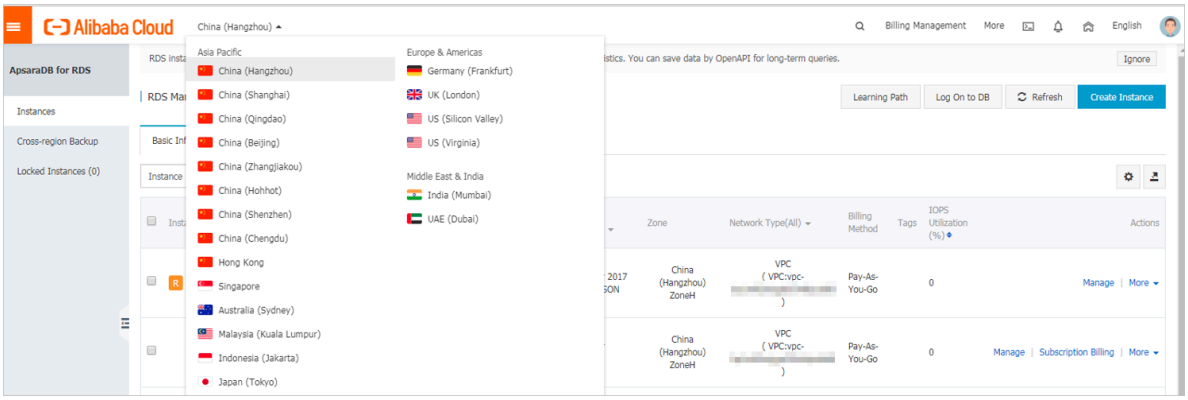
This topic describes how to delete tags from an RDS instance when you no longer need the tags or due to adjustments to the instance.

Limits

After you unbind a tag from an RDS instance, the tag is deleted if it is not bound to any other instance.

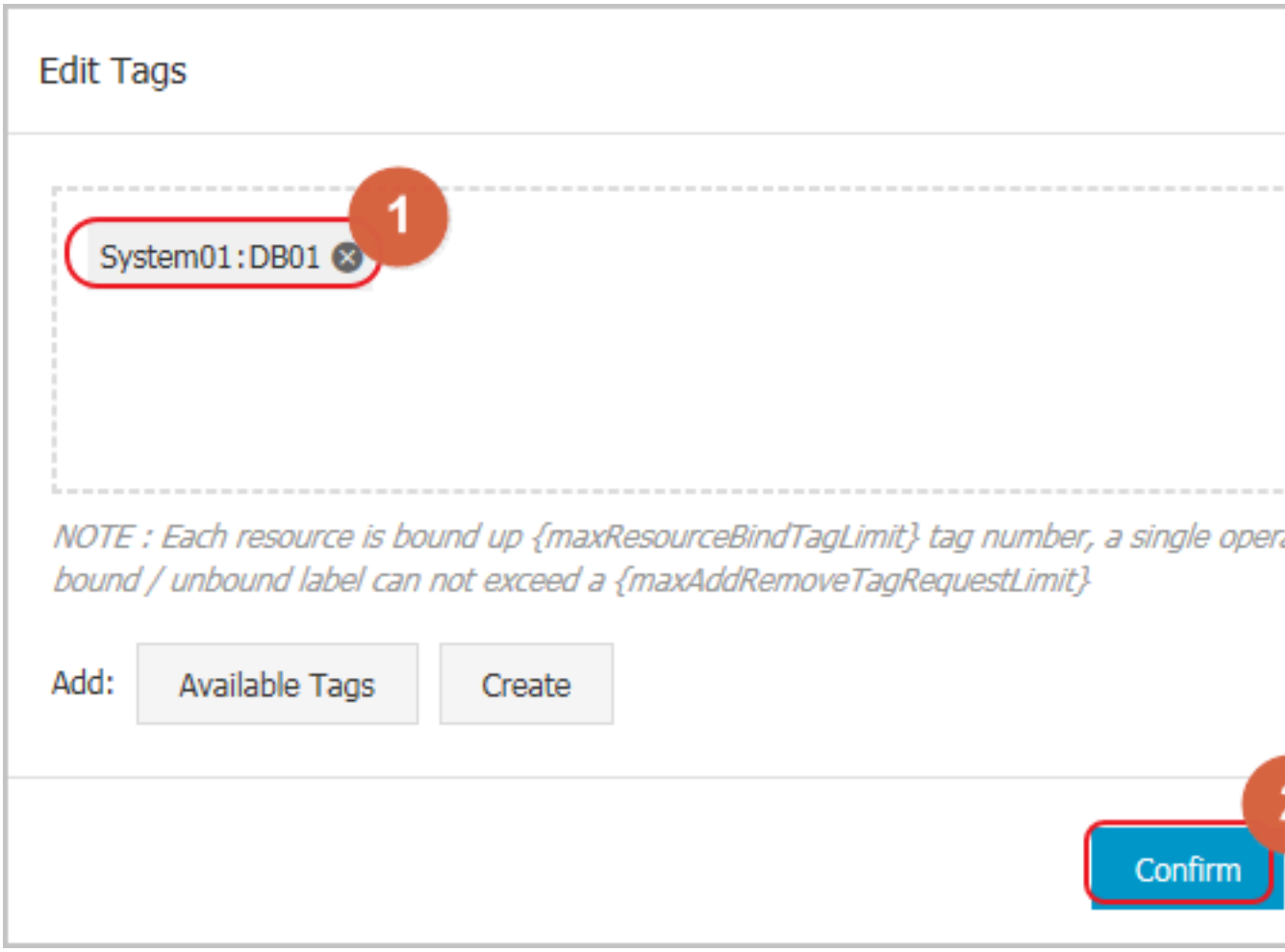
Procedure

1. **Log on to the *RDS console* and in the left-side navigation pane, click Instances.**
2. **In tne upper-left corner, select the region where the target RDS instance is located.**



3. **Find the target RDS instance and in the Actions column, choose More > Edit Tag.**

4. **Find the tag you want to delete, and click the X button following the tag.**



5. **Click Confirm.**

APIs

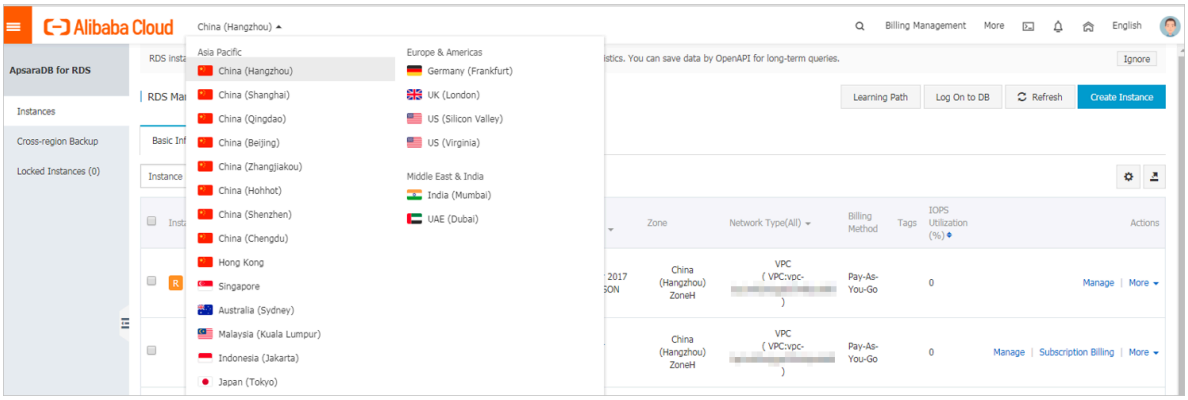
API	Description
#unique_209	**Used to unbind a tag from an RDS instance.**

20.3 Filter RDS instances by tag

This topic describes how to filter RDS instances by tag.

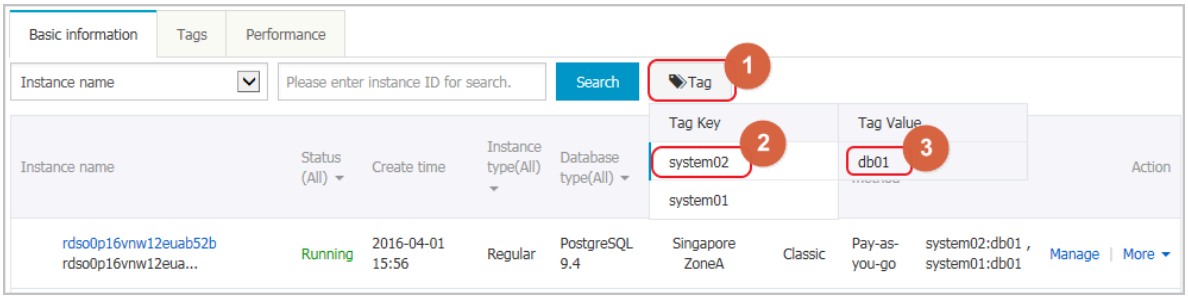
1. **Log on to the** *RDS console*.

2. In the upper-left corner, select the region where the target RDS instance is located.



3. On the Basic Information tab, click the Tag button next to Search and select a tag key and a tag value.

> **Note:**
> You can click the X button following the tag key to cancel the filter operation.



APIs

API	Description
DescribeTags	Used to query tags.

21 Appendixes

21.1 Commonly used SQL statements for MySQL

This topic lists some of the commonly used SQL statements.

For more information about the SQL statements including parameters and restrictions, see *MySQL 5.7 Reference Manual*.

Database-related SQL statements

Operation	Example
Create a database and designate a character set.	`create database db01 DEFAULT CHARACTER SET gbk COLLATE gbk_chinese_ci;`
Delete a database.	`drop database db01;`

Account-related SQL statementss

Note:
If an RDS instance has a premier account, the passwords of the other accounts under this instance cannot be changed by using the premier account. To change

the password of another account, you must delete this account and create a new one.

Operation	Example
Create an account.	`CREATE USER 'username'@'host' IDENTIFIED BY 'password';`
Delete an account.	`DROP USER 'username'@'host';`
Authorize the account.	`GRANT SELECT ON db01. * TO 'username'@'host';`

Query the created accounts in the database.	`SELECT user,host,password FROM mysql.user_view;`
	or
	`show grants for xxx`

Reclaim permissions.	• **Reclaim all permissions:**
	`REVOKE ALL PRIVILEGES,GRANT OPTION FROM 'username'@'host';`
	• **Reclaim specific permissions:**
	`REVOKE UPDATE ON *. * FROM 'username'@'host';`

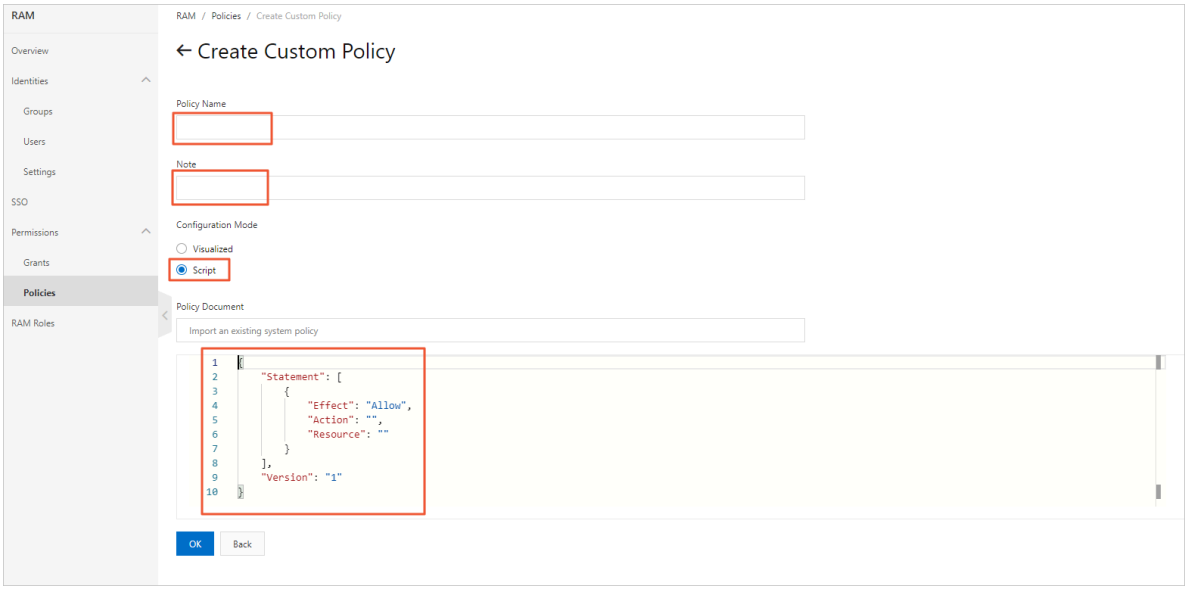
21.2 Grant backup file download permissions to a RAM user with only read-only permissions

This topic describes how to grant backup file download permissions to a RAM user who only has read-only permissions. For security purposes, a typical RAM user cannot download backup files.

Procedure

1. **Log on to the *RAM console*.**
2. **In the left-side navigation pane, choose Permissions > Policies.**

3. Click Create Policy and set the parameters.

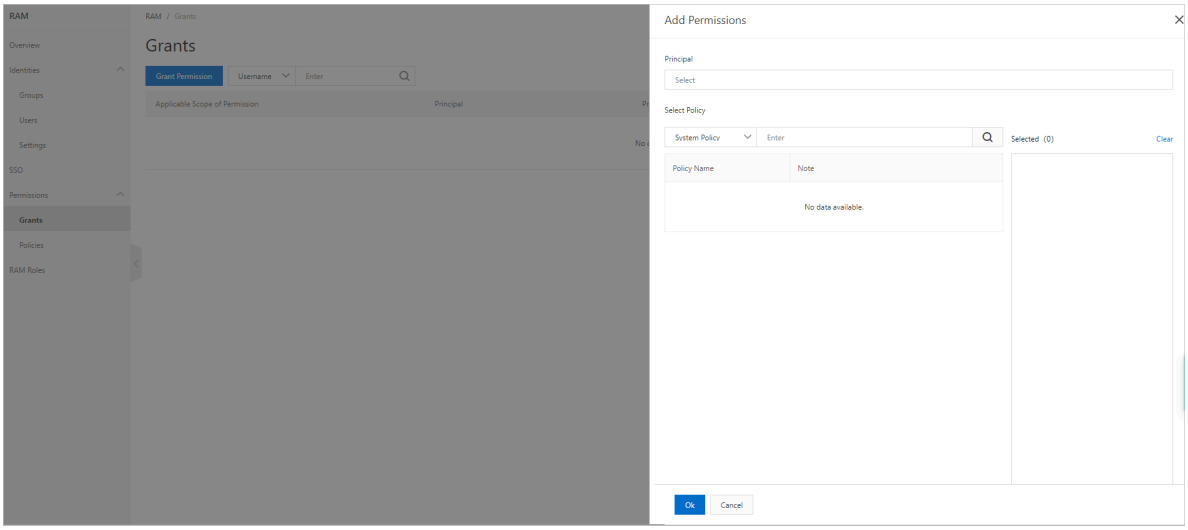


The policy content is as follows:

```
{
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "rds:Describe*",
                "rds:ModifyBackupPolicy"
            ],
            "Resource": "*"
        }
    ],
    "Version": "1"
}
```

4. Click OK.
5. In the left-side navigation pane, choose Permissions > Grants.

6. **Click Grant Permission and add the new permission policy to the target RAM user.**



7. **Click OK.**

21.3 Cached data persistence

ApsaraDB RDS can be used together with ApsaraDB Memcache and Redis to form storage solutions with high throughput and low delay. This document describes the cached data persistence solution based on the combined use of RDS and Memcache .

Background information

Compared with RDS, Memcache and Redis have the following features:

- **Quick response: The request delay of ApsaraDB Memcache and Redis is usually within several milliseconds.**
- **The cache area supports a higher Queries Per Second (QPS) than RDS.**

System requirements

- **bmemcached (with support for SASL extension) has been installed in the local environment or ECS.**

  bmemcached download address: Click *Here* to download.

  The bmemcached installation command is as follows:

  ```
  pip install python-binary-memcached
  ```

- **Python is used as an example. Python and pip must be installed in the local environment or ECS.**

Sample code

The following sample code realizes the combined use of ApsaraDB RDS and Memcache:

```
/usr/bin/env python
import bmemcached
Memcache_client = bmemcached.Client((‘ip:port’), ‘user’, ‘passwd’)
#Search for a value in ApsaraDB Memcache
res = os.client.get(‘test’)
if res is not None:
    return res #Return the value found
else:
    #Query RDS if the value is not found
    res = mysql_client.fetchone(sql)
     Memcache_client.put(‘test’, res) #Write cached data to ApsaraDB
for Memcache
    return res
```

21.4 Multi-structure data storage

OSS is a cloud storage service provided by Alibaba Cloud, featuring massive capacity, security, low cost, and high reliability. RDS can work with OSS to form multiple types of data storage solutions.

For example, when the business application is a forum and RDS works with OSS, resources such as registered users' images and post content images can be stored in OSS to reduce the storage pressure of RDS.

Sample code

OSS works with the RDS.

1. **Initialize OssAPI.**

```
from oss.oss_api import *
endpoint=”oss-cn-hangzhou.aliyuncs.com”
accessKeyId, accessKeySecret=”your id”,”your secret”
oss = OssAPI(endpoint, accessKeyId, accessKeySecret)
```

2. **Create a bucket.**

```
#Set the bucket to private-read-write
res = oss.create_bucket(bucket,"private")
print "%s\n%s" % (res.status, res.read())
```

3. **Upload an object.**

```
res = oss.put_object_from_file(bucket, object, "test.txt")
```

```
print "%s\n%s" % (res.status, res.getheaders())
```

4. **Obtain the corresponding object.**

```
res = oss.get_object_to_file(bucket, object, "/filepath/test.txt")
print "%s\n%s" % (res.status, res.getheaders())
```

In the ECS application code, RDS stores the ID of each user, and OSS stores the avatar resource of the user. The Python code is as follows:

```
/usr/bin/env python
from oss.oss_api import *
endpoint=”oss-cn-hangzhou.aliyuncs.com”
accessKeyId, accessKeySecret=”your id”,”your secret”
oss = OssAPI(endpoint, accessKeyId, accessKeySecret)
User_id = mysql_client.fetch_one (SQL) # Search for user_id in RDS
#Obtain and download the user avatar to the corresponding path
oss.get_object_to_file(bucket, object, your_path/user_id+’.png’)
#Process the uploaded user avatar
oss.put_object_from_file(bucket, object, your_path/user_id+’.png’)
```